A Survey on the Cryptographic Encryption Algorithms

Muhammad Faheem Mushtaq, Sapiee Jamel, Abdulkadir Hassan Disina, Zahraddeen A. Pindar, Nur Shafinaz Ahmad Shakir, Mustafa Mat Deris

Faculty of Computer Science and Information Technology,
Universiti Tun Hussein Onn Malaysia (UTHM),
86400, Parit Raja, Batu Pahat, Johor, Malaysia

***Abstract*—Security is the major concern when the sensitive information is stored and transferred across the internet where the information is no longer protected by physical boundaries. Cryptography is an essential, effective and efficient component to ensure the secure communication between the different entities by transferring unintelligible information and only the authorized recipient can be able to access the information. The right selection of cryptographic algorithm is important for secure communication that provides more security, accuracy and efficiency. In this paper, we examine the security aspects and processes involved in the design and implementation of most widely used symmetric encryption algorithms such as Data Encryption Standard (DES), Triple Data Encryption Standard (3DES), Blowfish, Advanced Encryption Standard (AES) and Hybrid Cubes Encryption Algorithm (HiSea). Furthermore, this paper evaluated and compared the performance of these encryption algorithms based on encryption and decryption time, throughput, key size, avalanche effect, memory, correlation assessment and entropy. Thus, amongst the existing cryptographic algorithm, we choose a suitable encryption algorithm based on different parameters that are best fit to the user requirements.**

Keywords*—Cryptography; encryption algorithms; Data Encryption Standard (DES); Triple Data Encryption Standard (3DES); Blowfish; Advanced Encryption Standard (AES); Hybrid Cubes Encryption Algorithm (HiSea)

I. INTRODUCTION

Security plays an important role to store information and transmit it across the undefined networks with secure manner. Hence, the secure communication is the basic requirement of every transaction over networks. Cryptography is an essential component for secure communication and transmission of information through security services like confidentiality, data integrity, access control, authentication and non-repudiation. It provides a way to protect sensitive information by transferring it into unintelligible and only the authorized receiver can be able to access this information by converting into the original text. The process to convert the plaintext into ciphertext with the key is called encryption process and to reverse the process of encryption is called decryption process. The design of cryptographic algorithms is secure and efficient, low cost, require small memory footprint, easy to implement and utilized on multiple platforms. The vast range of applications is developed to secure cryptographic algorithms using different mathematical process. It is quite difficult to develop fully secure encryption algorithm due to the challenges from cryptanalysts who continuously trying to access any available cryptographic systems [1]-[5]. The right selection of algorithms is important to achieve high-security requirements which protect the cryptographic components to cryptanalysis [6].

Cryptographic systems can be divided into deterministic and probabilistic encryption scheme [7]. Deterministic encryption scheme allows the plaintext is encrypted by using keys that always provide the same ciphertext, but the encryption process is repeated many times. In this scheme, every plaintext has one to one relationship with the keys and ciphertext otherwise it will produce more than one output of particular plaintext during the decryption process. Probabilistic Encryption Scheme shows the plaintext has different ciphertext with the different keys. The probabilistic encryption scheme is significantly secure than the deterministic encryption scheme because it makes difficult for a cryptanalyst to access any sensitive information regarding plaintext that is taken from ciphertext and corresponding key. Furthermore, the cryptographic algorithms can be further divided into two main categories like keyless cryptosystem and key-based cryptosystem as shown in Fig. 1. In the keyless cryptosystem, the relationship between the plaintext and ciphertext having a different version of the message is exclusively depend on the encryption algorithm [8]. The keyless cryptosystem is generally less secure than key-based systems because anyone can gain access to the algorithm will be able to decrypt every message that was encoded using keyless cryptosystem such as Caesar cipher [9]. The key-based cryptosystem can be further categories into symmetric key (secret key) encryption and asymmetric key (public key) encryption based on the type of security keys utilized for the encryption or decryption process [10]-[13]. The detail of the cryptosystems is explained as follows:

A. Symmetric Key Encryption

The symmetric key (secret key) encryption is employed similar key for the encryption and decryption of a message. Encryption and decryption keys are keeping secret and only known by authorized sender and recipient who want to communicate. The allocation of different keys to the different parties increases the overall message security. The strength of the symmetric key encryption is depending on the secrecy of encryption and decryption keys. The symmetric encryption algorithms can be classified into block and stream cipher on

This paper was partly sponsored by the Centre for Graduate Studies UTHM

the basis of the grouping of message bits [14], [15]. In a block cipher, a group of messages characters of a fixed size (a block) is encrypted all at once and sent to the receiver. Moreover, the block cipher can be further divided into binary and non-binary block cipher based on the final results of the message, keys and ciphertext. The message bit size for the binary block cipher is 64, 128, 192, and 256 and the non-binary block cipher has not defined the standard that depends on the cipher implementation.

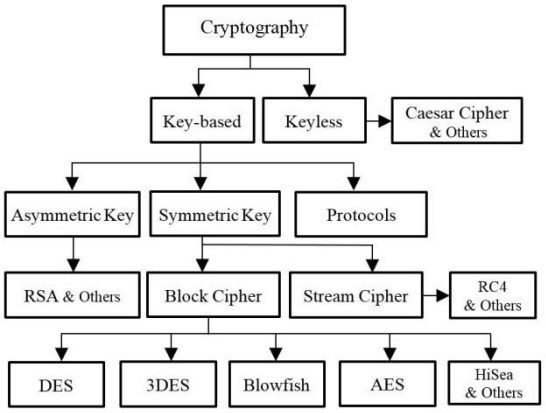

Fig. 1. Overview of the cryptographic encryption algorithms.

Symmetric key block cipher comprises the five main components: plaintext, encryption and decryption algorithm, ciphertext and key schedule algorithm as shown in Fig. 2. There are several symmetric key encryption algorithms such as DES [16], [17], 3DES [9], AES [18], [19], BLOWFISH [20], HiSea [21], RC4 [22], etc. The encryption process in symmetric block cipher converts the plaintext into ciphertext with the secret key that is generated from the key schedule algorithm. Similarly, the ciphertext is transferred to the appropriate recipient is decrypted using decryption process with the same key.

The block size for the stream cipher is one character and it is not more appropriate for software processing due to the key length as long the message [23], [24]. The working of the stream cipher is presented in following steps:

1) A single character of plaintext is combined with a single character from key stream to produce the single character of ciphertext.

2) The ciphertext character from Step 1 sent to the receiver.

3) Step 1 and Step 2 is repeated until the entire message has been sent.

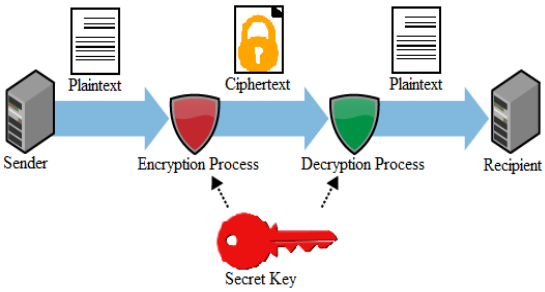

Fig. 2. Components of symmetric block cipher.

B. Asymmetric Key Encryption

The asymmetric key encryption is commonly referred to as public key encryption in which different keys are employed for the encryption and decryption of the message. The encryption key is also said as the public key and can be utilized to encrypt the message with the key. The decryption key is said to as secret or private key and can be used to decrypt the message. The strength of the asymmetric key encryption is utilized with digital signature then it can provide to the users through message authentication detection. The asymmetric encryption algorithm includes RSA [25], Diffie-Hellman algorithm [26], etc. The component of an asymmetric block cipher is shown in Fig. 3.

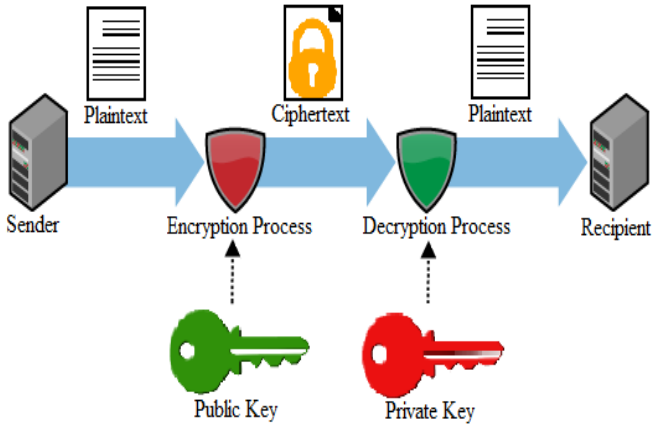

Fig. 3. Components of asymmetric block cipher.

C. Key Schedule Algorithm

Key schedule algorithm is employed to generate secret keys and plays an important role in the development of encryption and decryption key. The insignificant key generation algorithm generates weak keys that are used for encryption process can easily attack using brute force attack because cryptanalyst continuously trying all possible combinations to get original text using this attack [27]-[29]. All cryptographic algorithms follow the consideration of Advanced Encryption Standard (AES) that must support the key lengths include 128 bits, 192 bits and 256 bits [19]. The number of the round for that key length is 10, 12, 14 respectively and the round keys are taken from the cipher key using key schedule algorithm and utilized in the construction of block cipher. For the development of fully secure block cipher, the multiple numbers of rounds ensure the high diffusion and employed invertible transformation.

D. Shannon's Principles for Symmetric Block Cipher

Claude Shannon [30] proposed a set of five criteria for good ciphers is defined as follows:

1) In order to cipher a message, the degree of secrecy is required to determine the amount of labor. The value of information tends to decline over time, so additional computation labor is needed to protect the message secrecy for thousands of years that may not be secure in the perspective of information theory.

2) Cryptographic keys and encryption algorithms should

be free from complexity. Encryption algorithm should capable to encrypt any message using any key and the algorithm easy to understand.

3) Implementation of a cipher should be as simpler as possible.

4) Generation of error should be limited.

5) The size and storage required for the ciphertext message should be restricted. Make sure that from where it was executed, the size of the ciphertext should not exceed the size of plaintext under any circumstances.

From the historical perspective, it is interesting to note that these five criteria for good cipher are proposed earlier of the computer age and still they are perfectly valid. Furthermore, the Shannon's introduce the two principles of confusion and diffusion that are very important and closely related the functionality included in the development of secure encryption and decryption algorithms [30], [31]. The principle of confusion refers as the hides and complicating the relationship between the ciphertext and the keys (encryption or decryption key) as much as possible. It will help to prevent from cryptanalyst to predict the secret key using ciphertext. The principle of diffusion refers as the hides and complicating the relationship between the plaintext and ciphertext. It will ensure the small modification in the plaintext effects the unpredictable changes and create avalanched effect to the ciphertext. The relationship between the confusion and diffusion with cipher component is shown in Fig. 4.

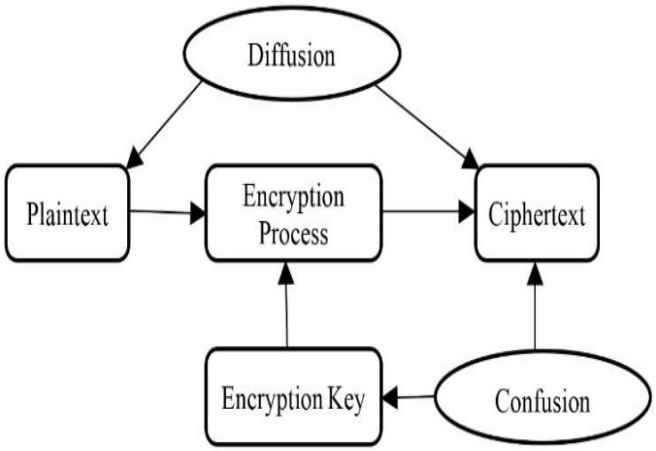

Fig. 4. Relationship between the confusion and diffusion.

E. Evaluation Parameters

To evaluate the efficiency and security, it is required to pass the execution test. Every encryption algorithm has some strength and weaknesses. In order to employ a secure encryption scheme to the applications, we need to evaluate the performance parameters [24], [32]-[35]. In this study, some of the evaluation parameters are discussed.

1) Encryption time: The time required to converts the plaintext into the ciphertext is said to an encryption time. The encryption time based on the message block size and the key size, and represented in milliseconds. It has direct impacts on the performance of the encryption algorithm. Every cryptographic algorithm required minimum encryption time, in order to make the encryption scheme responsive and fast.

2) Decryption time: The time required to recover the plaintext from ciphertext is said to decryption time. For the purpose of cryptographic algorithm fast and responsive, it is desirable that the decryption time less similar to the encryption time and it is also measured in milliseconds.

3) Memory used: Memory size depends on the implementation of different algorithms. The memory requirement depends on the key size, initialization vectors, and type of operations. It is more desirable that memory size should be small because it impacts on the cost of the system.

4) Throughput: For calculating the throughput of encryption algorithm by dividing the total block size (MegaByte) encrypted on the total encryption time. If the throughput value is increased, then the power consumption of algorithm is decreased.

5) Avalanche effect: It determines that there is any change in the plaintext, then the ciphertext will change significantly. In other words, we can say that it measures the dissimilarity between the plaintext and ciphertext changes. Avalanche effect can be measured using the hamming distance. If the high degree of diffusion required then the high avalanche effect is desirable. It reflects the performance of cryptographic algorithms and can be calculated by dividing the humming distance on the file size.

$$Avalanche = \frac{(Total\ number\ of\ bits - number\ of\ flip\ bits)}{Total\ number\ of\ bits} \text{x}100 \quad (1)$$

6) Entropy: The strength of overall implementation of the algorithm is estimated by using random matrix technique. Entropy is used to measure the randomness and uncertainty in the data. The relationship between the ciphertext and key becomes more complex with the high randomness. Encryption algorithms required high randomness in encrypting the plaintext, it results less or no dependency between the ciphertext and key. This property is referred as the confusion. A high degree of confusion is desirable that makes the difficulty for an attacker to guessed the entire set of information. To calculate Shannon's entropy test using the following equation:

$$H(X) = -\sum_{i=0}^{n-1} p(x_i) \log_b p(x_i) \quad (2)$$

7) The number of bits required for encoding optimally: This evaluation parameter defines the bandwidth required for transmission. An encrypted character or bits encoded with less number of bits, it will consume less storage and bandwidth. It also impacts on the cost of the system.

This paper explains the overview and performance factors involved during the design of symmetric encryption algorithms such as DES, 3DES, Blowfish, AES and HiSea. The main objectives of this research can be summarized as follows:

1) To review the existing encryption algorithms that explore what and how many parameters involved in the development of secure encryption technique.

2) To track the trends of research in this field.

3) To identify the significances of this area.

4) To present the existing performance evaluation in cryptographic schemes and suggest the best encryption scheme based on user requirement.

Firstly, we deeply review and compare the existing symmetric encryption algorithms based on security parameters. The selection of symmetric encryption algorithm instead of asymmetric encryption algorithm because its implementation is very fast, efficient, effective and simple to employ for encryption and decryption process. Furthermore, the AES is symmetric block cipher employed for encryption and decryption of message adopted by the United State of America [36]. Every cryptographic algorithm considered as approval with AES that required to fulfil the validation and execution time's test [19]. Later on, the performance analysis is based on the results of different researchers and addresses the fundamental aspects in the development of encryption algorithm that is based on encryption and decryption time, throughput, key size, avalanche effect, memory, correlation assessment and entropy for the selected cryptographic algorithm. Finally, the best suitable cryptographic algorithm is chosen based on different parameters for further research and future directions are also explored.

The remaining paper is organized as follows: Section II discusses some cryptographic encryption algorithms which include the overview of existing symmetric encryption algorithms. Section III explains the results and analysis of encryption algorithms that are discussed in the previous section. Section IV includes the conclusion and future directions of this research.

II. Cryptographic Encryption Algorithms

This section explains the review of existing encryption algorithms that are used to conclude the better encryption scheme based on different parameters.

A. Data Encryption Standard (DES)

DES is the earliest symmetric encryption algorithm developed by IBM in 1972 and adopted in 1977 as Federal Information Processing Standard (FIPS) by the National Bureau of Standard (NBS). The NBS is currently the National Institute of Standards and Technology (NIST) that evaluate and implement the standard encryption algorithm. It includes 64 bits key that contains 56 bits are directly utilized by the algorithm as key bits and are randomly generated. The remaining 8 bits that are not used by algorithm because it is used for the error detection as set to make a parity of each 8-bit byte [17], [37], [38]. DES utilized the one secret key for encryption and decryption process and key length is 56 bits and performs the encryption of message using the 64 bits block size. Similarly, the decryption process on a 64 bits ciphertext by using the same 56 bits key to produce the original 64 bits block of the message is shown in Fig. 5. The DES algorithm processes the 64 bits input with an initial permutation, 16 rounds of the key and the final permutation.

The DES algorithm structure is based on Feistel function that divided the block into two halves. The function (f) based on the four stages such as expansion, key mixing, substitution and permutation. The number of rounds applies for the DES is 16 used for the processing to encrypt the message.

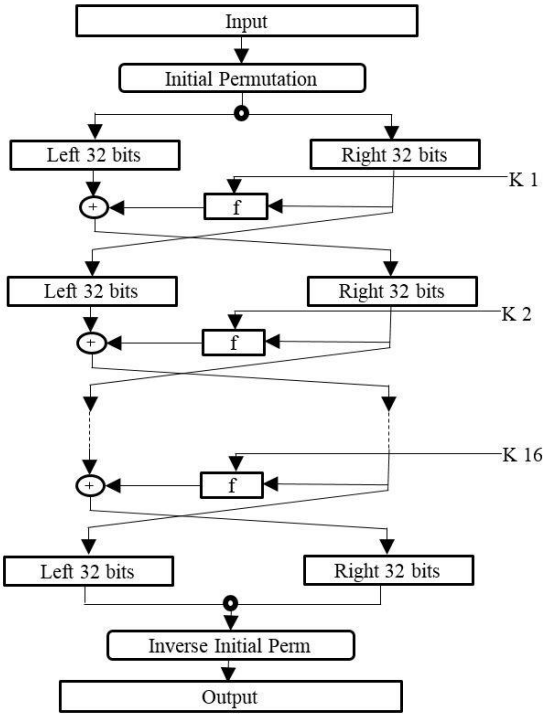

Fig. 5. Data Encryption Standard (DES) Algorithm.

The output after the 16 round consists of 64 bits that are the function of the input message and the key. DES mostly used in the banking industry, commercial and military secret information sharing purpose. Security is the major concern in DES because it uses the 56 bits key (2^{56}) or 7.2 x 10^{16} keys and cryptanalysts are trying to crack an encrypted message by key exhaustion. Brute force attack is possible through parallel machines of more than 2000 nodes with each node that has capabilities of key search 50 million keys/sec [39]. DES is cracked in 1998 by using Electronic Freedom Foundation constructed device within 22 hours due to the less number of key length and is highly susceptible to the linear cryptanalysis attacks.

B. Triple Data Encryption Standard (3DES)

Triple Data Encryption Standard (3DES) referred as Triple Data Encryption Algorithm (TDEA) that was firstly proposed by IBM in 1998 and standardized in ANSI X9.17 and ISO 8732. 3DES was appeared as the replacement of DES due to the improvement in the key length and applies the DES algorithm to the three times in each data block. The 56 bits key length of DES algorithm was generally adequate earlier when the algorithm designed but as the computation power increases then the brute force attack is feasible. On the other hand, 3DES provides a very simple method by the increment of key length instead of design a complete block cipher and it protects against the brute force attack. A brute force attack continuously trying every possibility of accessing keys until

the original message is obtained. Table 1 demonstrated four key sizes that show how long it required for the various key spaces [9], [40]. The DES employed the 56 bits key size and 3DES utilized the 168 bits key size.

The key length for the 3DES is 112 bits and 168 bits, the number of rounds 48 and the block size is 64 bits [41]. The purpose of this algorithm is to increase the security with longer key length, so it is challenging for the cryptanalyst to predict the pattern and attacks become rapidly impractical. The Key size, Number of keys, Time required at 1 Decryption/μs and Time required at 10^6 Decryption/μs is represented as Ks, Nk, Tr1D, Tr10^6D respectively.

TABLE I AVERAGE TIME REQUIRED FOR EXHAUSTIVE KEY SEARCH

Ks	**Nk**	**Tr1D**	**Tr10^6D**
32-bits	$2^{32} = 4.3 \times 10^9$	$2^{31}\mu s = 35.8$ minutes	2.15 millisecond
56-bits	$2^{56} = 7.2 \times 10^{16}$	$2^{55}\mu s = 1142$ years	10.01 hours
128-bits	$2^{128} = 3.4 \times 10^{38}$	$2^{127}\mu s = 5.4 \times 10^{24}$ years	5.4×10^{18} years
168-bits	$2^{168} = 3.7 \times 10^{50}$	$2^{167}\mu s = 5.9 \times 10^{36}$ years	5.9×10^{30} years
26 characters (permu)	$26! = 4 \times 10^{26}$	$2 \times 10^{26}\mu s = 6.4 \times 10^{12}$ years	6.4×10^6 years

The main advantage of the 3DES algorithm is three times secure having key size 2^{168} (use keys as a combination or each level with different keys size) as compared to DES algorithm having key size 2^{56}, that's why 3DES algorithm is preferred as compared to the DES algorithm. Moreover, it provides adequate security to the information but the problem with that it consumes more time in encryption process as compare to DES. The encryption algorithm of 3DES is presented as follows:

$$C = Encrypt_{K3}(Decrypt_{K2}(Encrypt_{K1}(P))) \quad (3)$$

and decryption algorithm of 3DES is given as follows:

$$P = Decrypt_{K3}(Encrypt_{K2}(Decrypt_{K1}(C))) \quad (4)$$

Where C represented the ciphertext, P represented the plaintext and K1, K2, K3 represent the keys.

The overview and attraction of 3DES over next few years can be defined in two ways [9]. Firstly, it overcomes the vulnerability of brute force attack of the DES by using 168 bits key size. Secondly, the encryption algorithm of 3DES is similar as in DES due to that more analysis than another algorithm over long time period. Moreover, this algorithm didn't find any effective cryptanalysis attack rather than brute force. If we analyze in term of security, then 3DES appears as a suitable choice for the standard encryption algorithm in future decades. The major drawback of the 3DES algorithm is that it is slow in software because DES was designed in 1972 in hardware implementation with no efficient software. 3DES algorithm has three more times more rounds, that's why it is correspondingly slow. The second drawback of DES and 3DES is that it uses 64 bits block size and for the demand of more security and efficiency, the large block size is desirable.

C. Advanced Encryption Standard (AES)

The NIST announced a call for the candidates of a cipher to implement a new encryption standard in 1997 because of the need for high security and efficiency, it's time to replace the existing DES and 3DES encryption algorithm with new AES. All candidates of ciphers submitted its proposal by 1998 and finalized in 2000. Finally, Rijndael was selected as the AES out of 15 candidates. Rijndael is developed by Vincent Rijmen and Joan Daemen in 2001. US government is employed AES to protect sensitive information and implemented across the world for data encryption purpose in form of software and hardware. AES appears as the recent generation block cipher and significantly increases in the block size up to 128 bits with the key sizes 128 bits, 192 bits and 256 bits. The number of rounds set with respective key size is the 10, 12, 14 for the 128 bits, 192 bits, 256 bits, respectively [9], [42], [43]. The number of AES parameters based on the key length mentioned in Table 2. The parameters Key size, Block size, Number of rounds, Round key size, and expanded key size are represented as Ks, Bs, Nr, Rks, Eks, respectively.

AES was designed with the following characteristics:

- Compactness of code and speed on the large range of platforms.
- Simple design.
- Protection against all known attacks.

The data blocks are used as the array of bytes and represented in a matrix that is referred as the state array which changed in every step of encryption and decryption process. Each round follows some steps during encryption process to complete each round until 'n'. After the final step, the state array is transferred into output matrix [18], [19], [44]. The steps for each round consist of four layers i.e. substitute byte, shift rows, mix column and add round key is shown in Fig. 6. In the first layer, S-box of order 8 is applied to each byte. For the linear mixing, the second and third layers are used. In these layers, the columns are mixed, and rows of the array are shifted. The subkey bytes are XORed with every byte of the array in the fourth layer. The round operation is done iteratively that is based on the key size. The decryption process has also the similar operation and same sequences of transformations as with the encryption, but it employed in the reverse order. The transformation is an inv-substitute byte, inv-shift rows, inv-mix columns and adds round key that assigns the key schedule form as identical for encryption and decryption process. All operation of AES can be combined into XOR operation and a lookup table, so the implementation can be very efficient and fast.

TABLE II ADVANCED ENCRYPTION STANDARD PARAMETERS

Ks (words/bytes/ bits)	**Bs (words/bytes/ bits)**	**Nr**	**Rks (words/bytes/bits)**	**Eks (words/ bytes)**
4/16/128	4/16/128	10	4/16/128	44/176
6/24/192	4/16/128	12	4/16/128	52/208
8/32/256	4/16/128	14	4/16/128	60/240

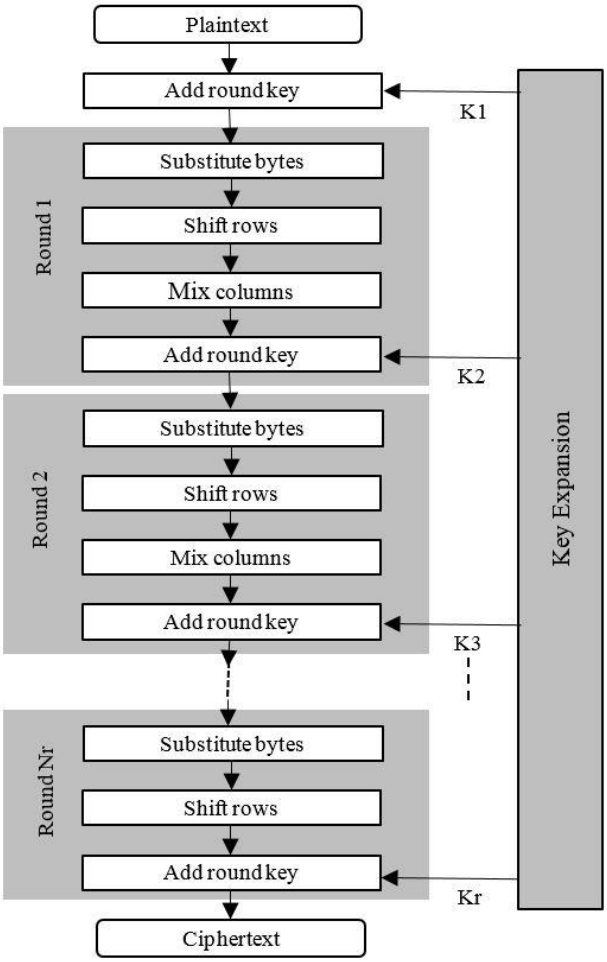

Fig. 6. Advanced Encryption Standard (AES) Algorithm.

D. Blowfish

Blowfish is symmetric block cipher based on the Feistel function that is effectively used for encryption and decryption process. It was introduced by one of most leading cryptologists Bruce Schneier in 1993. Most of the encryption algorithms are not available for the public and most of them are protected by patent. Blowfish is fast, license free, unpatented, freely available and alternative for existing encryption algorithms. It uses the key length range up to 32-448 and 64 bits block. Blowfish algorithm employed 16 rounds for the encryption process. Blowfish is a Feistel structure that consists of 16 rounds shown in Fig. 7. This algorithm considerably analyzed and with the instance of time, it gains popularity as a robust block cipher [38]. Like the other ciphers, this algorithm also effectively used in VLSI hardware and can be optimized in software application [16], [45]. The input as a plaintext is 64 bits data E.

Divide the data E into two halves of 32 bits: EL, ER

For i = 1 to 16:

EL = EL XOR EPi

ER = Fn (EL) XOR ER

Swap EL and ER

Next i

Swap EL and ER (Undo the last swap.)

ER = ER XOR EP17

EL = EL XOR EP18

Recombine EL and ER

Function Fn is represented as follows:

Divide EL into four 8 bits quarters: w, x, y, and z

Fn(EL) = ((S1, w + S2, x mod 2^{32}) XOR S3, y) + S4, z mod 2^{32}.

The decryption process of Blowfish algorithm is similar as encryption process, except that EP1, EP2, ..., EP18 are employed in the reverse order. Blowfish primarily utilized four S-boxes instead of the one S-box to prevent similarity between the different bytes when the input is equal to the 32 bits input to the function Fn is bytewise permutation with other input of 32 bits [46]. This algorithm used one S-box in each process, so four different outputs are generated a non-trivial permutation of each output. The design of four S-box seems more secure, faster and easy to program. The function that joins the output of four S-boxes is fast that would be XOR the four output values with mix addition of mod 2^{32}. The repetition of addition in each round and all XOR operations end with an addition because the final result is combined with XOR to the RE.

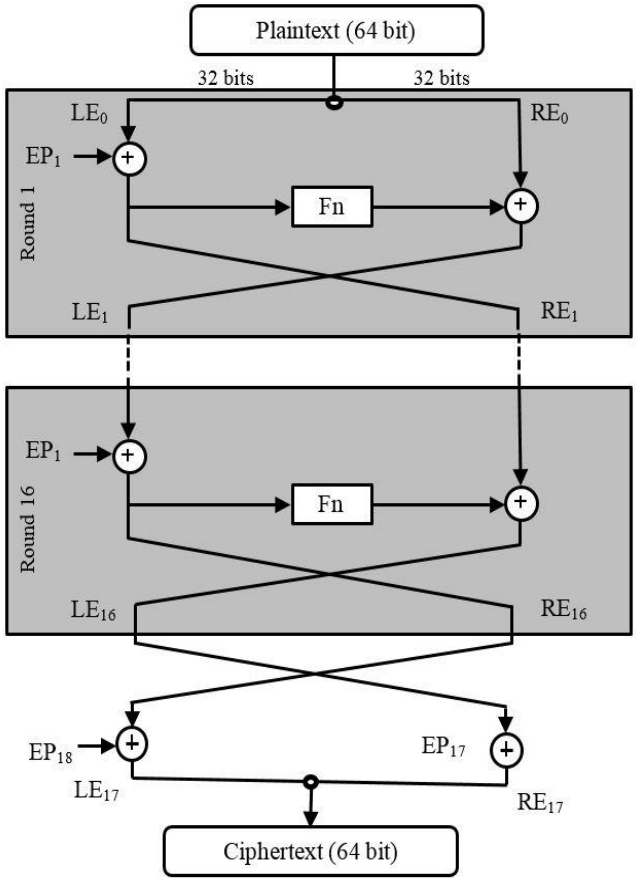

Fig. 7. Blowfish Encryption Algorithm.

Blowfish algorithm needed more processing time because it depends on the key size. The subkey generation process increases the complexity that protects from brute force attack and provides better security than existing encryption techniques. Moreover, the use of a large number of weak keys will damage the reliability of Blowfish [39]. It also utilized 64 bits block, but the larger block size is more desirable.

E. Hybrid Cubes Encryption Algorithm (HiSea)

Hybrid Cube Encryption Algorithm (HiSea) is the symmetric non-binary block cipher because the encryption and decryption key, plaintext, ciphertext and internal operation in the encryption or decryption process that is based on the integer numbers. HiSea encryption algorithm is developed by Sapiee Jamel in 2011. The plaintext size for the encryption process is the decimal ASCII characters of 64 bytes. Hybrid Cube (HC) is generated based on the inner matrix multiplication of the layers between the two Magic Cubes (MC) [47]. HC of order 4x4 matrix $H_{i,j}$, $i \in \{1, 2, 879\}$ and $j \in \{1, 2, 3, 4\}$ is defined as follows:

$$H_{i,j} = MC_{i,j} \times MC_{i+1,j} \quad (5)$$

where the $MC_{i,j}$ is a j^{th} layer of i^{th} magic cubes.

Let us consider the HC 1 is generated through the inner matrix multiplication of MC 1 layer with {x=1, 2, 3, 4} coordinates and MC 2 layer having coordinates {x=1, 2, 3, 4}. Similarly, HC 2 is generated with the inner matrix multiplication of MC 2 and 3, and so on. A new cube structure HC is generated by using the layers of MC where the layer entries lie between the set of integers {1, 2, 3, … 4096}. All possible combination of HC layer entries can be utilized to increase complexity in the design of encryption and decryption algorithms [48], [49]. The overall design of the HiSea in which the plaintext, keys and ciphertext in encryption process are formatted into order 4 matrix is shown in Fig. 8. The encryption algorithm used the following steps:

1) The plaintext is format as 64 characters into 64 Extended ASCII codes and four matrices of Plaintext is represented as P1-P4. The intermediate result (P1′) for P1 is used in the encrypting process of P2. The intermediate result (P2′) for P2 is used in the encrypting process of P3. This process is repeated for P4. The major reason for integrating this method to ensure any change made in P1 will reflect into another ciphertext. The process of diffusion is performing on the initial stage to increase complexity in the ciphertext.

2) P1 is mixed with Initial Matrix (IM), P2, P3 and P4 that generate the temporary ciphertext called P1′. P1′ is then added with the session Key 1 (K1). The Function MixRow and MixCol are used to create diffusion in Ciphertext 1 (C1).

3) P2 is mixed with P1′ to produce P2′. This plaintext is then added with session Key 2 (K2). The results derive through MixRow and MixCol that create diffusion in Ciphertext 2 (C2).

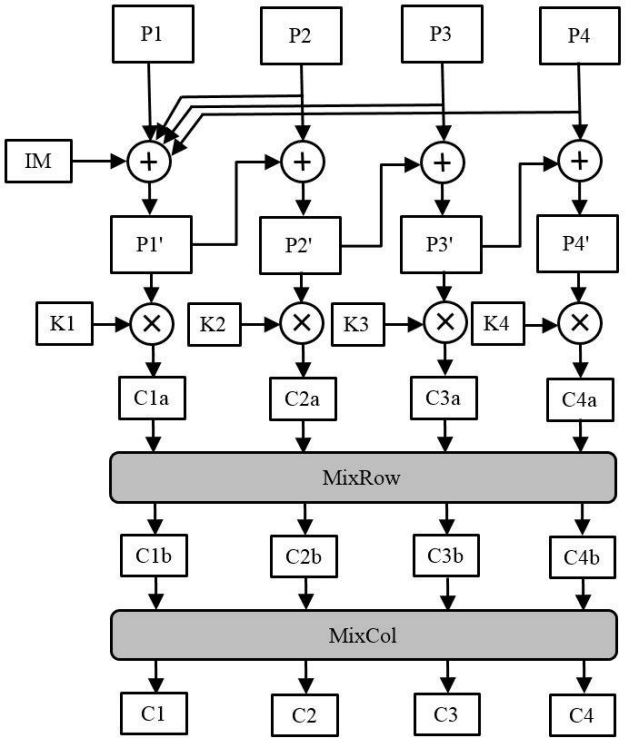

Fig. 8. Hybrid Cube Encryption Algorithm (HiSea).

4) Repeat step 3 with P3 and P4 to generate Ciphertext 3 (C3) and Ciphertext 4 (C4)

HiSea is computationally secure and has a large key space $10^{153.6}$ keys that make the brute force attack difficult or time-consuming [21], [27]. Furthermore, the comparison of different encryption algorithm is presented in Table 3.

III. Results and Discussion

This section explains the performance analysis based on the results of different researchers and addresses the security aspects in the development of encryption algorithm based on the evaluation parameters. Moreover, there is a number of studies that assembling up-to-date development and improvement in this field. Some researchers have a major focus on surveying about the cryptographic algorithms and their performance evaluation. Generally, the performance of the block cipher depends on the block size and key length. The large block size will make the algorithm faster because a large portion of data will be encrypted in the single execution cycle. Similarly, the small block of data required more execution cycle that increase the overall execution time. On the other hand, the large key size will affect the algorithm performance because all key bits are involved in algorithm execution that makes the slower performance. However, the large key length brings the algorithm more security and provides more protection against cryptanalyst. Moreover, the importance of performance evaluation is to determine the software and hardware related best configuration setting, allowing the assessment that which one algorithm setting is more efficiently and effectively solve the problem.

TABLE III COMPARATIVE ANALYSIS OF SYMMETRIC ENCRYPTION ALGORITHM

Algorithms/Parameters	DES	3DES	AES	Blowfish	HiSea
Published	1977	1998	2001	1993	2011
Developed by	IBM	IBM	Vincent Rijmen, Joan Daeman	Bruce Schneier	Sapiee Jamel
Algorithm Structure	Feistel	Feistel	Substitution-Permutation	Feistel	Substitution-Permutation
Block cipher	Binary	Binary	Binary	Binary	Non-Binary
Key Length	56 bits	112 bits, 168 bits	128 bits, 192 bits and 256	32 – 448 bits	1 – 4096 set of integers
Flexibility or Modification	No	YES, Extended from 56 to 168 bits	YES, 256 key size is multiple of 64	YES, 64-448 key size in multiple of 32	No
Number of Rounds	16	48	10, 12, 14	16	4
Block size	64 bits	64 bits	128 bits	64 bits	64 characters
Throughput	Lower than AES	Lower than DES	Lower than Blowfish	High	Lower than AES
Level of Security	Adequate security	Adequate security	Excellent security	Excellent security	Highly secure
Encryption Speed	slow	Very slow	Fast	Fast	Moderate
Effectiveness	Slow in both software and hardware	Slow in software	Effective in both software and hardware	Efficient in software	Efficient in software
Attacks	Brute force attack	Brute force attack, Known plaintext, Chosen plaintext	Side channel attack	Dictionary attack	Not yet

A performance comparison of symmetric encryption algorithms based on the execution time using Electronic Codebook (ECB) and Cipher Feedback (CFB) modes was considered [38]. They used different data size in bytes (20527, 36002, 45911, 59862, 69646, 137325, 158959, 166364, 191383 and 232398) for both modes and apply test on DES, 3DES, AES and Blowfish. Firstly, they execute the test using ECB mode on Pentium II having 266 MHz and Pentium 4 having 2.4 GHz machine, respectively. The average execution time (in seconds) of both machines and the comparison of the average execution time is given in Table 4 and Fig. 9.

TABLE IV AVERAGE EXECUTION TIME OF ENCRYPTION ALGORITHM IN ECB MODE

Algorithms/Machine	DES	3DES	AES	Blowfish
Pentium 2	134	383	228	108
Pentium 4	14	42	21	11

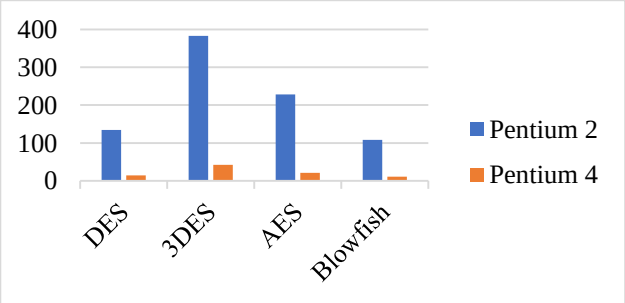

Fig. 9. Average execution time of algorithms in ECB mode.

It shows that the execution time in the encryption process of Blowfish is faster than the rest of the techniques but 3DES appear to be the slow in term of execution time using ECB mode. Moreover, the same data size is applied to find the execution time (seconds) in CFB mode shown in Table 5 and comparison of the results in Fig. 10.

TABLE V AVERAGE EXECUTION TIME OF ENCRYPTION ALGORITHM IN CFB MODE

Algorithms/Machine	DES	3DES	AES	Blowfish
Pentium 2	1015	2909	3551	812
Pentium 4	106	328	328	86

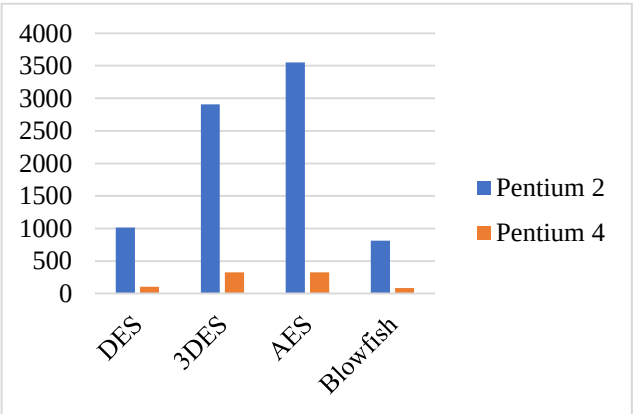

Fig. 10. Average execution time of algorithms in CFB mode.

The average execution time in encryption time shows that the Blowfish is faster than the other encryption technique using the CFB mode. Also, it is noted that the 3DES takes more encryption time as compare to DES due to the triple key size. Meanwhile, the performance evaluation of symmetric encryption algorithms that are based on different blocks size, key size, data types, encryption/decryption time and power consumption is explained [50], [51]. This paper evaluates the encryption algorithms such as DES, 3DES, AES and Blowfish and calculates the throughput by changing the block size. They used different block size in Kbytes (49, 59, 100, 247, 321, 694, 899, 963, 5345.28 and 7710.336) for the encryption and decryption algorithms. The execution is done on laptop IV and CPU 2.4 GHz. The throughput value of encryption and decryption process is shown in Table 6 and Fig. 11.

TABLE VI THROUGHPUT OF ENCRYPTION AND DECRYPTION ALGORITHM

Algorithm/ Process	DES	3DES	AES	Blowfish
Encryption	4.01	3.45	4.174	25.892
Decryption	6.347	5.665	6.452	18.72

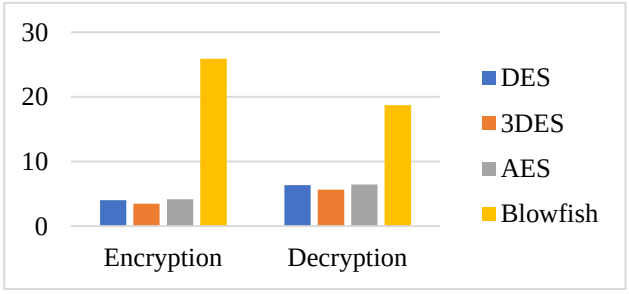

Fig. 11. Throughput of encryption and decryption algorithms (Megabyte/Sec).

The experimental result shows that the throughput value of Blowfish is better in encryption and decryption process than the other algorithms. The basic terminology is that the throughput value increases, then the power consumption will be decreased. We found that the AES performance is better than DES and 3DES. Moreover, the performance evaluation of DES and Blowfish is based on execution speed using different memory sizes and explain the relationship between the function memory size and run speed [16]. In this paper, performance is estimated on PC Pentium (R) 4, 3.00 GHz and run program 109 times to encrypt plaintext of 256 characters. The memory size is from 96M to 992M as illustrated in Table 7 and Fig. 12.

TABLE VII COMPARE THE RUN TIME (µS) BETWEEN DES AND BLOWFISH

Memory size/ Algorithm	96	224	352	480	608	736	992
DES	1.1528	0.8370	0.8330	0.8463	0.8350	0.8440	0.8445
Blowfish	0.1503	0.1373	0.1234	0.1234	0.1250	0.1245	0.1245

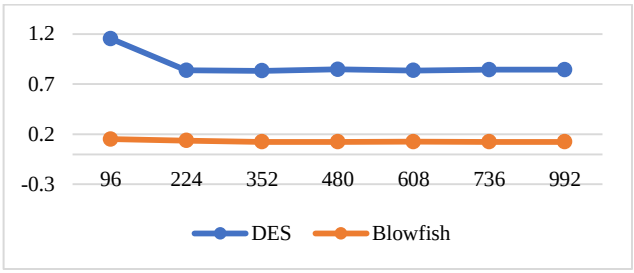

Fig. 12. Execution speed of DES and Blowfish.

The results demonstrate that the blowfish execution speed is faster than DES, but it consumes more memory to initialize the Subkey and S-box than the DES.

Meanwhile, the performance evaluation of the DES and AES is based on the parameters such as memory, simulation time and avalanche effect on Pentium dual-core T4300, 2.0 GHz with RAM 2GB [1]. The analysis of DES and AES based on different parameters is shown in Table 8 and Fig. 13.

AES shows significantly high avalanche effect than DES by changing the one bit in plaintext keep the constant key and variation of bits from 83 to 81. Also, it shows AES required less memory and execution time. So, AES is a better choice where the less memory is required. Meanwhile, the performance evaluation of DES and AES is based on the encryption time by using Intel Pentium processor 2.34 GHz and 1GB RAM [11]. Different size of files is used to evaluate the performance as demonstrated in Table 9.

The results show that the encryption time of AES is less than DES. So, it means that AES performance is much better than the DES as shown in Fig. 14.

TABLE VIII COMPARISON OF DES AND AES BASED ON AVALANCHE EFFECT, REQUIRED MEMORY AND EXECUTION TIME

Algorithm	Variation of 1 bit in plaintext having constant key	Variation of 1 bit in key having constant plaintext	Required memory	Execution Time
DES	43	41	43.3	0.32
AES	83	81	10.2	0.0304

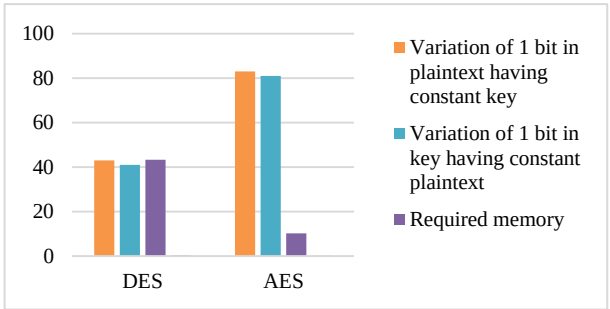

Fig. 13. Execution speed of DES and Blowfish.

TABLE IX PERFORMANCE EVALUTION BASED ON ENCYPTION TIME

File Size (KB)	32	126	200	246	280
DES	0.27	0.83	1.19	1.44	1.67
AES	0.15	0.46	0.72	0.95	1.12

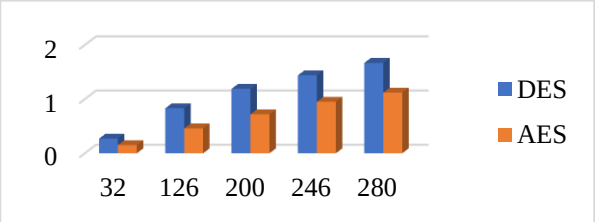

Fig. 14. Performance evaluation based on encryption time.

The evaluation of HiSea encryption algorithm is based on randomness between ciphertext and key, and the correlation between the message and ciphertext [48]. The simulation was performed on HP 2530, core 2 duo, 2.13 GHz and RAM 2GB. The Ciphertext (C1-C4), Session Key (SK), Initial Matrix (IM) and Correlation Assessment is illustrated in Table 10.

TABLE X ENTROPY AND CORRELATION ASSESMENT FOR HISEA

IM	SK	C1	C2	C3	C4	CA
0.8199	0.8632	0.9999	0.9998	0.9998	0.9995	0.0440

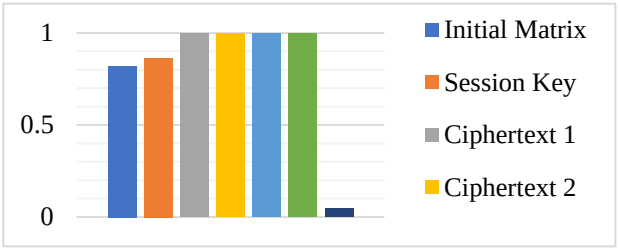

Fig. 15. Entropy (IM, SK, C1-C4) and correlation test.

The entropy results show that the keys generated through hybrid cubes are 0.8632 or 86.32% random and the initial matrix that is used to add plaintext 1 during the encryption process is 81.99% random demonstrated in Fig. 15. The keys used to generate a ciphertext are more than 99% random which means that ciphertext (C1-C4) blocks are almost random and hide the relationship between the key and ciphertext. The value of correlation test on HiSea is 0.0440 which means that there is no correlation exists between the message and ciphertext pairs. Furthermore, the performance analysis of AES (128, 192, and 256), DES, 3DES and Blowfish are based on the average response time with different data size of 1MB, 3MB, 7MB and 10MB using the laptop 2.4 GHz is shown in Table 11.

TABLE XI PERFORMANCE EVALUATION BASED ON AVERAGE RESPONSE TIME

File Size (MB)	DES	3DES	Blowfish	AES 128	AES 192	AES 256
1	0.14	0.39	0.08	0.12	0.14	0.15
3	0.38	1.08	0.22	0.33	0.37	0.43
7	0.99	2.71	0.51	0.79	0.91	1.03
10	1.34	3.63	0.71	1.10	1.25	1.41

The results demonstrate that the Blowfish takes less time to encrypt the specified file size than other encryption algorithms as shown in Fig. 16. AES-128 and AES-192 show good response time than DES and 3DES but AES-256 takes more time than DES to encrypt the file sizes [52]. 3DES shows more response time to encrypt the file and provide less performance than the other encryption algorithms.

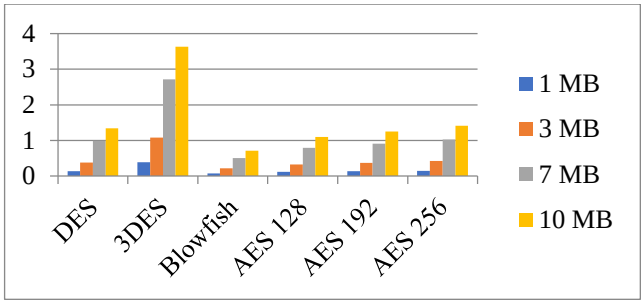

Fig. 16. Evaluation based on average response time.

In the end, overall results of encryption algorithms are presented based on the different evaluation parameters shown in Table 12. All cryptographic algorithms are depending on the block size, key and number of rounds. Generally, the algorithms must consider the security requirements such as computational resources availability, the application's requirements, and the distribution of secure key. In order to apply appropriate encryption algorithm for the applications, we must have knowledge about the strength, weakness, and performance based on different parameters. Blowfish is appeared as fast encryption scheme in terms of execution time, throughput and runtime that is better than DES, 3DES, AES and HiSea. An analysis based on brute force and correlation assessment shows that the HiSea encryption and decryption key are suitable in the development of secure non-binary block cipher. AES algorithm has excellent avalanche effect and high in execution time and performed better as compare to DES, 3DES. The problem with the AES algorithm is that it requires a significant amount of resources and power. The 3DES show the low performance in the following parameters because it uses 64 bits block size and 168 bits keys that have no more modification from DES only the three-time the key size, but it effects on throughput, encryption and decryption time. Also, DES and 3DES software are not more efficient. The demand for more efficiency and security is that it required the large block size and key size, smallest the memory and resources are desirable.

IV. CONCLUSION AND FUTURE WORK

A comprehensive review based on the cryptographic algorithm for the security of data has been performed in this paper. The detailed summary of a symmetric block ciphers such as DES, 3DES, AES, Blowfish and HiSea along with different design methodologies have been presented. The demonstration of results and discussion about these algorithms are mainly focused on evaluation parameter like encryption and decryption time, memory, avalanche effect, throughput, correlation assessment and entropy because these parameters show a more security, confidentiality, integrity, and reliability for secure communication. Based on the performance evaluation, the results of Blowfish, AES and HiSea provide more security based on the resources availability. Blowfish is the best option in those applications where the memory and

encryption/decryption time is the major factor and it is efficient in software. However, AES can be evaluated based on the avalanche effect that shows excellent performance and HiSea show good performance in term of entropy and correlation assessment. So, we conclude that the AES and HiSea can be employed in those applications where integrity and confidentiality is the highest priority.

HiSea provides great strengthened to the encryption algorithm because the entropy of encryption keys is 99% random. However, based on the comparison between the symmetric block ciphers, the Blowfish is a best suitable candidate for security and it has the potential for further development due to a significant advantage in memory, encryption and decryption time, throughput and efficient encryption design. Based on the above study, this research analyzes that there is a need to develop the hybrid encryption algorithm which combines different encryption algorithms based on all suitable parameters that are used to enhance the overall security of the encryption techniques.

TABLE XII PERFORMANCE EVALUTION BASED ON DIFFERENT PARAMETERS

Algorithm	Evaluation Parameters	Nadeem et al. (2005)	Elminaam et al. (2008, 2009)	Nie et al. (2009)	Jamel et al. (2011)	Mandal et al. (2012)	Silva et al. (2016)	Faiqa et al. (2017)
DES	Encryption time	**				*	**	*
	Throughput		**					
	Run time test			*				
	Memory			***		*		
	Avalanche effect					*		
3DES	Encryption time	*					*	
	Throughput		*					
AES	Encryption time	***				****	***	****
	Throughput		***					
	Avalanche effect					****		
	Memory					***		
Blowfish	Encryption time	****					****	
	Throughput		****					
	Run time test			****				
	Memory space			**				
HiSea	Entropy				****			
	Correlation assessment				****			

* = Low; ** = Medium; *** = High; **** = Excellent

ACKNOWLEDGMENT

This paper was partly sponsored by the Centre for Graduate Studies, Universiti Tun Hussein Onn Malaysia (UTHM), Johor, Malaysia. Moreover, the authors would like to thank the Ministry of Higher Education Malaysia for supporting this research under Fundamental Research Grant Scheme (FRGS), Vote No. 1642.

REFERENCES

[1] K. Mandal, C. Parakash, and A. Tiwari, "Performance evaluation of cryptographic algorithms: DES and AES," in Proceeding of the IEEE Students' Conference on Electrical, Electronics and Computer Science: Innovation for Humanity (SCEECS), 2012.

[2] M. Ebrahim, S. Khan, and U. bin Khalid, "Symmetric algorithm survey: A comparative analysis," International Journal of Computer Applications, vol. 61, no. 20, pp. 12–19, 2013.

[3] N. Kumar and P. Chaudhary, "Performance evaluation of encryption/decryption mechanisms to enhance data security," Indian Journal of Science and Technology, vol. 9, no. 20, 2016.

[4] Disina, A. H., Pindar, Z. A., & Jamel, S., "Enhanced caeser cipher to exclude repetition and withstand frequency cryptanalysis," Journal of Network and Information Security, 2015.

[5] V. V Palagushin and A. D. Khomonenko, "Evaluation of cryptographic primitives security based on proximity to the latin square," in Proceeding of the IEEE 18th conference of fruct association, pp. 266–271, 2016.

[6] S. H. Jamel and M. M. Deris, "Diffusive primitives in the design of modern cryptographic algorithms," in proceedings of the International Conference on Computer and Communication Engineering (ICCCE08): Global Links for Human Development, pp. 707–710, 2008.

[7] S. Goldwasser and M. Bellare, Lecture Notes on Cryptography, Cambridge, Massachusetts, 2008.

[8] A. Kaushik, M. Barnela, and A. Kumar, "Keyless user defined optimal security encryption," International Journal of Computer and Electrical Engineering, vol. 4, no. 2, pp. 2–6, 2012.

[9] W. Stallings, Cryptography and network security: principles and practices. Prentice Hall, 2005.

[10] M. Stamp, Information Security: Principles and Practice. John Wiley & Sons, 2011.

[11] F. Maqsood, M. M. Ali, M. Ahmed, and M. A. Shah, "Cryptography: A comparative analysis for modern techniques," International Journal of Advanced Computer Science and Applications, vol. 8, no. 6, pp. 442–448, 2017.

[12] S. Ahmad, K. M. R. Alam, H. Rahman, and S. Tamura, "A comparison between symmetric and asymmetric key encryption algorithm based decryption mixnets," in Proceedings of the IEEE International Conference on Networking Systems and Security, 2015.

[13] E. Fujisaki and T. Okamoto, "Secure integration of asymmetric and symmetric encryption schemes," Journal of Cryptology, vol. 26, no. 1,

pp. 80–101, 2013.

[14] A. M. Alshahrani and S. Walker, "Implement a novel symmetric block cipher algorithm," International Journal on Cryptography and Information Security, vol. 4, no. 4, pp. 1–11, 2014.

[15] M. Dworkin, "Recommendation for block cipher modes of operation," NIST Spec. Publ. 800-38B, 2005.

[16] T. Nie and T. Zhang, "A study of DES and Blowfish encryption algorithm," in Proceedings of 10th IEEE Region Annual International Conference TENCON, pp. 1–4, 2009.

[17] M. E. Smid and D. K. Branstad, "Data Encryption Standard: past and future," Proceedings of the IEEE, vol. 76, no. 5, pp. 550–559, 1988.

[18] J. Daemen, V. Rijmen, and K. U. Leuven, AES Proposal: Rijndael. (NIST), National Institute of Standards, 1999.

[19] N. I. of Standards-(NIST), Advanced Encryption Standard (AES). Federal Information Processing Standards Publication197, 2001.

[20] B. Schneier, "Description of a new variable-length key, 64-bit block cipher (Blowfish)," in Proceedings of the Fast Software Encryption: Cambridge Security Workshop Cambridge, U. K., pp. 191–204, 1994.

[21] S. Jamel, M. M. Deris, I. T. R. Yanto, and T. Herawan, "The hybrid cubes encryption algorithm (HiSea)," Communications in Computer and Information Science, Springer-Verlag Berlin Heidelberg, vol. 154, pp. 191–200, 2011.

[22] W. Stallings, "The RC4 stream encryption algorithm," in Cryptography and network security, 2005.

[23] S. B. Sasi, N. Sivanandam, and Emeritus, "A survey on cryptography using optimization algorithms in WSNs," Indian Journal of Science and Technology, vol. 8, no. 3, pp. 216–221, 2015.

[24] S. Jamel, "The hybrid cubes encryption algorithm (HiSea)," Ph.D Thesis, Univ. Tun Hussein Onn Malaysia (UTHM), Johor, Malaysia, pp. 1–138, 2012.

[25] S. Burnett and S. Paine, RSA Security's Official Guide to Cryptography. McGraw-Hill, 2001.

[26] A. Escala, G. Herold, and C. Ràfols, "An algebraic framework for Diffie - Hellman assumptions," Journal of Cryptology, 2015.

[27] M. F. Mushtaq, S. Jamel, K. M. Mohamad, S. A. A. Khalid, and M. M. Deris, "Key generation technique based on triangular coordinate extraction for hybrid cubes," Journal of Telecommunication, Electronic and Computer Engineering (JTEC), vol. 9, no. 3-4, pp. 195-200, 2017.

[28] A. H. Disina, S. Jamel, M. Aamir, Z. A. Pindar, M. M. Deris, and K. M. Mohamad, "A key scheduling algorithm based on dynamic quasigroup string transformation and All-Or-Nothing key derivation function," Journal of Telecommunication, Electronic and Computer Engineering, vol. 9, no. 3–5, pp. 1–6, 2017.

[29] A. H. Disina, S. Jamel, Z. A. Pindar, and M. M. Deris, "All-or-nothing key derivation function based on quasigroup string," in proceeding of IEEE International Conference on Information Science and Security (ICISS), pp. 6–10, 2016.

[30] C. E. Shannon, "Communication theory of secrecy systems," Bell System Technical Journal, vol. 28, no. 4, pp. 656–715, 1949.

[31] H. X. Mel and D. M. Baker., Cryptography decrypted. Addison-Wesley, 2001.

[32] P. Patil, P. Narayankar, D. G. Narayan, and S. M. Meena, "A comprehensive evaluation of cryptographic algorithms: DES, 3DES, AES, RSA and Blowfish," Procedia Computer Science, vol. 78, pp. 617–624, 2016.

[33] N. Jorstad and T. Landgrave, "Cryptographic algorithm metrics," 20th National Information Systems Security, 1997.

[34] M. F. Mushtaq, U. Akram, I. Khan, S. N. Khan, A. Shahzad, and A. Ullah, "Cloud computing environment and security challenges: A review," International Journal of Advanced Computer Science and Applications, vol. 8, no. 10, pp. 183–195, 2017.

[35] D. Salama, A. Minaam, H. M. Abdual-kader, and M. M. Hadhoud, "Evaluating the effects of symmetric cryptography algorithms on power consumption for different data types," International Journal of Network Security, vol. 11, no. 2, pp. 78–87, 2010.

[36] ISO/IEC-18033-3, "Information technology - Security techniques-encryption algorithms - Part 3: Block ciphers," International Standard ISO / IEC, 2005.

[37] N. I. of S. and T. NIST, "Data Encryption Standard (DES)," Federal Information Processing Standards Publication (FIPS PUB 46-3), vol. 25, no. 10, pp. 1–22, 1999.

[38] A. Nadeem and M. Y. Javed, "A performance comparison of data encryption algorithms," International Conference on Information and Communication Technologies, pp. 84–89, 2005.

[39] Z. Hercigonja, D. Gimnazija, and C. Varazdin, "Comparative analysis of cryptographic algorithms and advanced cryptographic algorithms," International Journal of Digital Technology & Economy, vol. 1, no. 2, pp. 1–8, 2016.

[40] W. Stallings, Cryptography and Network Security: Principles and Practice, 5th ed. Prentice Hall Press, 2010.

[41] E. B. William C. Barker, "Recommendation for the triple data encryption algorithm (TDEA) block cipher," NIST Special Publication 800-67, 2012.

[42] J. Nechvatal, E. Barker, L. Bassham, W. Burr, M. Dworkin, J. Foti, and E. Roback, "Report on the development of the advanced encryption standard (AES)," National Institute of Standards and Technology, pp. 1–116, 2000.

[43] L. May, M. Henricksen, W. Millan, G. Carter, and E. Dawson, "Strengthening the key schedule of the AES," Proceedings of the 7th Australian Conference on Information Security and Privacy, pp. 226–240, 2002.

[44] J. Daemen and V. Rijmen, The Design of Rijndael - The Advanced Encryption Standard. Springer-Verlag Berlin Heidelberg, New York, 2002.

[45] B. Schneier, "Description of a new variable-length key, 64-bit block cipher (Blowfish)," Proc. Fast Softw. Encryption Cambridge Secur. Work. Cambridge, U. K., pp. 191–204, 1994.

[46] S. Manku and K. Vasanth, "Blowfish encryption algorithm for information security," ARPN Journal of Engineering and Applied Sciences, vol. 10, no. 10, pp. 4717–4719, 2015.

[47] S. Jamel, T. Herawan, and M. M. Deris, "A cryptographic algorithm based on hybrid cubes," Computational Science and Its Applications ICCSA, vol. 6019, pp. 175–187, 2010.

[48] S. Jamel, M. M. Deris, I. Tri, R. Yanto, and T. Herawan, "HiSea: A non binary toy cipher," Journal of Computing, vol. 3, no. 6, pp. 20–27, 2011.

[49] M. F. Mushtaq, S. Jamel, and M. M. Deris, "Triangular Coordinate Extraction (TCE) for hybrid cubes," Journal of Engineering and Applied Sciences, vol. 12, no. 8, pp. 2164–2169, 2017.

[50] D. Elminaam, H. M. A. Kader, and M. M. Hadhoud, "Performance evaluation of symmetric encryption algorithms," International Journal of Computer Science and Network Security, vol. 8, no. 12, pp. 280–286, 2008.

[51] D. S. A. Elminaam, H. M. A. Kader, and M. M. Hadhoud, "Evaluating the performance of symmetric encryption algorithms," International Journal of Computer Theory and Engineering, vol. 10, no. 3, pp. 343–351, 2009.

[52] N. B. F. Silva, D. F. Pigatto, P. S. Martins, and K. R. L. J. C. Branco, "Case studies of performance evaluation of cryptographic algorithms for an embedded system and a general purpose computer," Journal of Network and Computer Applications, vol. 60, pp. 130–143, 2016.