Research on Various Cryptography Techniques

Yahia Alemami, Mohamad Afendee Mohamed, Saleh Atiewi

ABSTRACT--- Security plays a critical position in preserving information privacy and secrecy. Many encryption strategies are available to protect data during transmission or storage. These encryption methods vary in terms of strength, speed, and resource consumption (CPU usage, memory, and power). This study aims to present the most popular and interesting algorithms currently in use.

Index Terms — Cryptography, data security, public key, resource consumption, secret key.

I. INTRODUCTION

Security is an important in protecting data against intruders. One of the most important methods for ensuring data secrecy is cryptography. Cryptography is secret writing for data security protection. Well-hidden data cannot easily be read, modified or fabricated [1]. Cryptography protects crucial data via changing it into unclear data that can only be accessed via authorized receivers, who then converts the uncertain data into the original textual content. The process of changing original text into unclear text (ciphertext) with a certain key referred to as encryption, and the opposite of encryption process is referred to as decryption process.

Privacy and security management present challenges to e-exam. An e-exam database requires security and reliability. Thus, an e-exam user's identity must be established. Computerized exams are prone to significant problems such as leaks, attackers and so on. One solution is to encrypt the questions inside the database. Encryption is the conversion of plaintext to text that is not clear.

The two fundamental techniques for encrypting data are "symmetric cryptography," which entails the usage of the same key to encrypt/ decode information; and "asymmetric cryptography," which makes use of public and private keys to encrypt/ decode information.

Examples of symmetric algorithms are Data Encryption Standard (DES), Triple-DES (3DES), Blowfish, and Advanced Encryption Standard (AES). The most well-known asymmetric algorithms are RSA and ELGAMAL Schema.

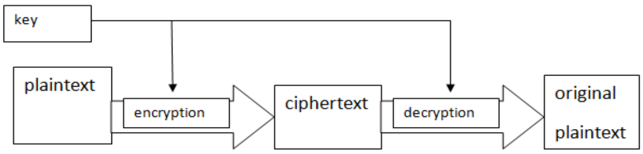

Fig. 1. Symmetric cryptosystem

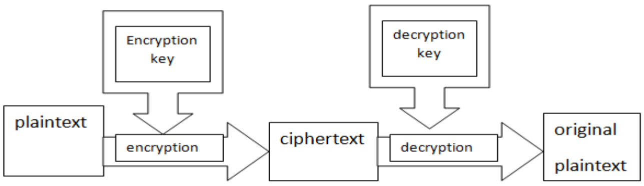

Fig. 2: Asymmetric cryptosystem

A. Common Terms Used in Cryptography

- Plaintext: The original and understandable text. As an instance, 'Y' needs to transmit a "Computer" message to 'Z'. Here, "Computer" is the plaintext or the original message.
- Ciphertext: The text that cannot be understood by way of anybody or a gibberish text, example "A@$&J9."
- Encryption: A process of changing clear text into unclear text. The manner of encipherment needs an encipherment algorithm and a key. Encipherment occurs on the sender side.
- Decryption: A reverse method of encode. It is a manner of converting ciphertext into plaintext.
- Key: A key is character, number, or a special character. It is used at the time of encipherment on the original text and at the time of decode on the ciphertext.

B. Purpose of Cryptography

- Authentication: The potential of a system to test the identity of the sender.
- Confidentiality: Information transmitted ought to be accessed handiest by using legal parties and not through anyone else.
- Integrity: Only the authorized parties are permitted to alter on transmitted information.
- Non-repudiation: Is the guarantee that someone cannot deny the validity of something.
- Access Control: Just the authorized persons are capable to get right of entry to the given information.

C. Evaluation Parameters

Each encryption algorithm presents strengths and weaknesses in terms of their parameters. Some parameters that determine encryption performance are described as follows.

1) Encryption time: Measured in milliseconds, depend on the data block length and key length. It directly influences the performance of the encryption algorithm. The performance of an algorithm is regarded as advanced when the encryption time is rapid.

Revised Manuscript Received on July 10, 2019.

Yahia Alemami, Faculty of Informatics and Computing, Universiti Sultan Zainal Abidin, Terengganu, Malaysia.
Mohamad Afendee Mohamed, Faculty of Informatics and Computing, Universiti Sultan Zainal Abidin, Terengganu, Malaysia.
Saleh Atiewi, Computer Science, Al Hussein Bin Talal University College of IT, Ma'an, Jordan.

Retrieval Number: B10690782S319/19©BEIESP
DOI : 10.35940/ijrte.B1069.0782S319

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

2) Decryption time: The time period to regain the original text from ciphertext; it is also measured in milliseconds. The performance of an algorithm is regarded as superior when the decryption time is rapid.
3) Memory used: A low memory usage is desirable because it affects system cost.
4) Throughput: Is computed through way of dividing the whole encoded block size on the entire encode time. The power consumption of the algorithm will decrease, if the throughput cost increases [2].

II. ENCRYPTION ALGORITHMS

This part clarifies numerous encryption algorithms to identify the best encryption schemes on the basis of various parameters.

A. Simplified Data Encryption Standard (S-DES)

The steps of the S-DES algorithm described as follows:

1) S-DES Key Generation: S-DES relies upon on the using of a shared key that consist of 10- bit and share it among both sender and receiver. Pair of 8-bit sub keys are generated (K1, K2) from this key for use in specific stages of encipherment and decipherment algorithms as presented in Fig. 3 [3].

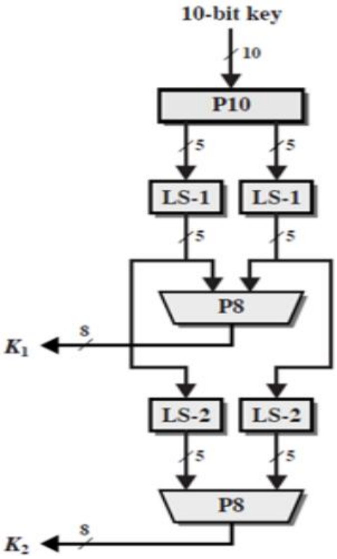

Fig. 3: Key generation for S-DES [3]

Both keys are coming through using the functions P10, Shift and P8 where P10 and P8 are as follows:

$$P10 = [3, 5, 2, 7, 4, 10, 1, 9, 8, 6]$$

$$P8 = [6, 3, 7, 4, 8, 5, 10, 9]$$

2) Initial and Final Permutations: The input to the algorithm is an eight-bit block of original text, which is initially permuted using the IP function with IP given as follows:

$$IP = [2, 6, 3, 1, 4, 8, 5, 7]$$

At the end of the algorithm, the inverse permutation, IP^{-1} is applied; here, IP^{-1} is given as follows:

$$IP^{-1} = [4, 1, 3, 5, 7, 2, 8, 6]$$

3) Function fK: This function can be explained as follows: The 8-bit input to fK is divided into left and right inputs, each of which contains 4 bits for use in the following equation: fK(L, R) = (L (XOR) F(R, SK), R) where SK is a subkey and XOR is the bit-by-bit exclusive-OR function. The initial process is an expansion/permutation process.

$$E/P = [4, 1, 2, 3, 2, 3, 4, 1]$$

The fundamental 4 bits are fed inside the S-box S0 to supply a 2-bit outcome, and the residual 4 bits are fed inside S1 to provide another 2-bit outcome. These boxes are described as follows:

$$S0 = \begin{array}{c|cccc|} & 0 & 1 & 2 & 3 \\ 0 & 1 & 0 & 3 & 2 \\ 1 & 3 & 2 & 1 & 0 \\ 2 & 0 & 2 & 1 & 3 \\ 3 & 3 & 1 & 3 & 2 \end{array} \qquad S1 = \begin{array}{c|cccc|} & 0 & 1 & 2 & 3 \\ 0 & 0 & 1 & 2 & 3 \\ 1 & 2 & 0 & 1 & 3 \\ 2 & 3 & 0 & 1 & 0 \\ 3 & 2 & 1 & 0 & 3 \end{array}$$

The S-boxes work as follows: The result of preceding steps indicates that the 1st and 4th enter bits are taken into consideration a two-bit number that specifies a row of the S-box, and the 2nd and 3rd input bits determine a column of the S-box. S0 is used by the left nibble, and S1 is used by the right nibble. The following 4 bits generated through S0 and S1 undergo further permutation as follows:

$$P4 = [2, 4, 3, 1]$$

Both Fig. 4 and 5 illustrate the S-DES.

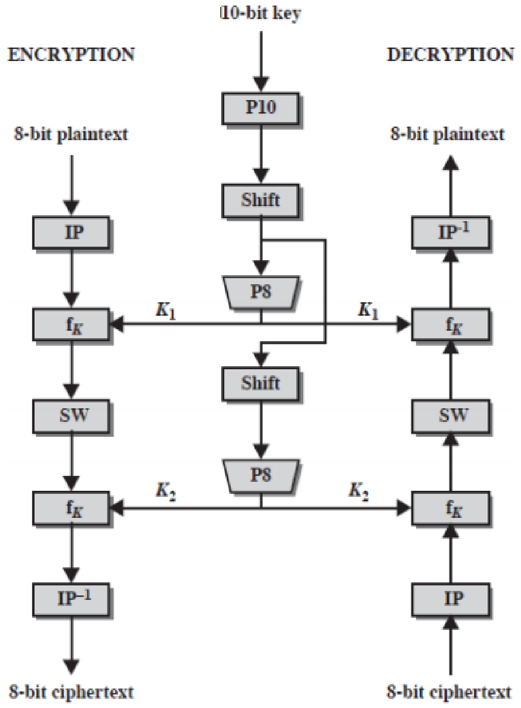

Fig. 4: S-DES scheme [4]

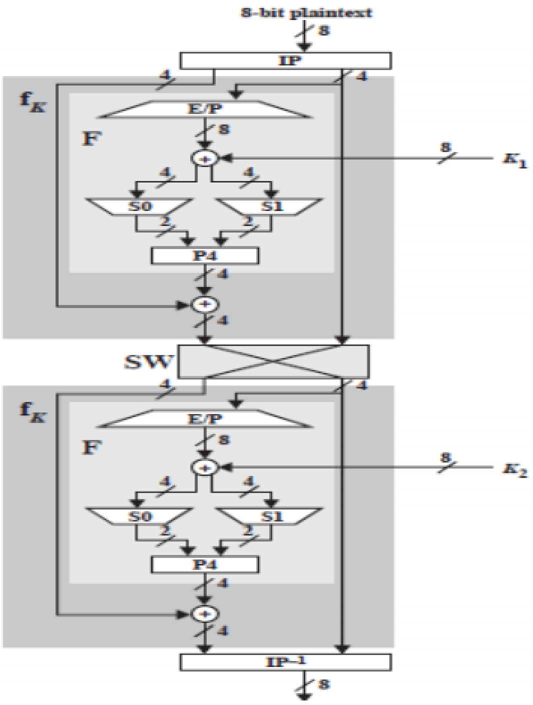

Fig. 5: S-DES encryption detail [4]

B. DES

DES is the earliest symmetric encipherment algorithm introduced in 1972 by International Business Machines Corporation and in 1977 it has been Agreed as Federal Information Processing Standard through the National Bureau of Standard [2]. It comprises the same steps as S-DES and processes a 64-bit input with a preliminary permutation. In DES, the number of rounds is 16 while in S-DES is two rounds also the S-DES uses 8 bits for input.

C. Triple DES (3DES)

3DES was suggested by IBM (International Business Machines Corporation) in 1998. A substitute for DES, 3DES shows improved key size and applies the DES algorithm 3 rounds in each data block. The key length for the 3DES is 112 and 168 bits, the number of rounds is 48 and the block length is 64 bits [2]. This algorithm aims to increase protection and security through its longer key size relative to DES. However, it is more time consuming than DES is when applied to the encryption process.

D. Blowfish

Blowfish is a type of symmetric block cipher generated by B. Schneier in 1993. Blowfish is fast algorithm, license free, and unpatented. It uses a key length in the range of 32–448 and a sixty-four-bit block. The Blowfish algorithm makes use 16 round for the encipherment procedure Fig. 6. Blowfish ordinarily makes use of 4 S-boxes rather than of one S-box. It requires additional processing time because it relies on key length, however it provides strong safety [2].

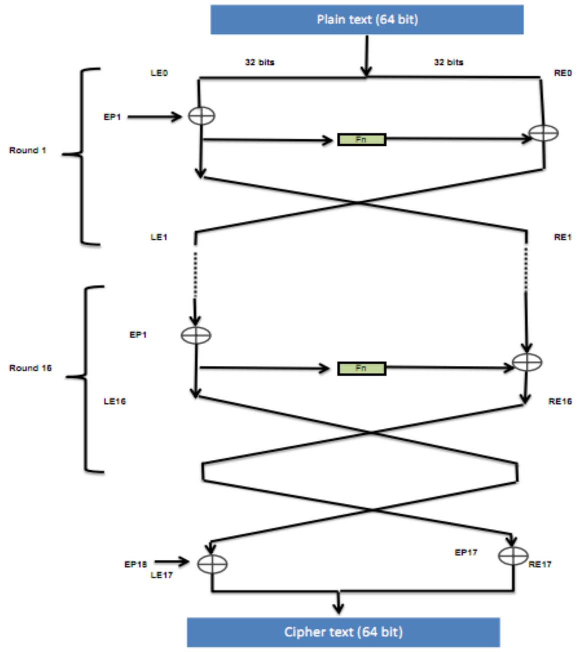

Fig. 6: Blowfish encryption algorithm [2]

E. Advanced Encryption Standard (AES)

AES was deployment by the National Institute of Standards and Technology in 2001; it is also called “Rijndael” [5].

AES is a block cipher with a block size of 128 bits. The key length can be 128, 192 or 256 bits. Encipherment includes ten rounds of processing for 128-bit keys, 12 rounds for 192-bit keys and 14 rounds for 256-bit keys. The algorithm is called AES-128, AES-192 or AES-256 relying on the key size [5]. The steps for every round include of 4 layers, particularly, replacement byte, shift rows, blend column and add round key as appear in Fig. 7 [2].

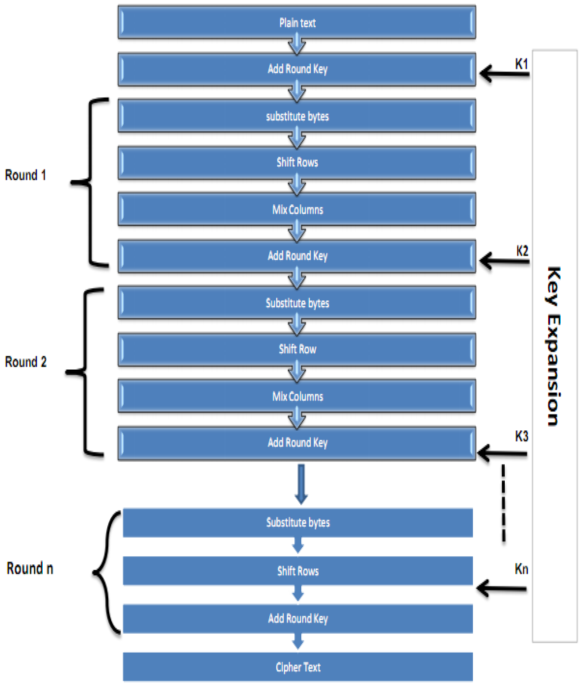

Fig. 7: AES algorithm [2]

III. LITERATURE REVIEW & RESULTS

A secure socket layer (SSL) channel guarantees that all information passed between the web server and the browser remain integral and secret.

Security is likewise applied to e-learning which includes e-exams, whose systems require authentication, privacy, encryption and confidentiality [6].

The Diffie–Hellman key is used for encipherment and decipherment using RSA algorithms. In addition, RSA and Diffie–Hellman are described to be sufficiently strong for commercial purposes. The Diffie–Hellman key can be used with RSA, DES, AES and elliptical curve cryptography to encipherment and decipherment messages [7].

The most popular encipherment and decipherment algorithms namely AES, DES, 3DES and Blowfish are set side by side in this work. The simulation results show that 3DES has the best performance with Electronic Codebook (ECB) and Cipher Blocker Chaining (CBC) amongst all the encryption algorithms used. A performance assessment of selected symmetric encipherment algorithms namely AES, DES, 3DES, Blowfish, RC2 and RC4 is also executed. The increase in key size length results in battery and time consumption [8].

Several studies have analyzed and surveyed the regions of encryption and decryption using DES, AES and RSA algorithms. In the future, these algorithms are anticipated to be implemented for secure and enhanced communication [9].

A methodology for encryption and decryption using ASCII algorithms is proposed. This new methodology is effective, fast, secure and reliable.

This algorithm is performed as follows: Any random number is selected and a subset is created with starting and ending numbers; a modulus is then selected, followed by the division of the subset by mode; the remainder is taken as the substitution array [10].

The use of a simple data protection model based on the AES algorithm and Diffie–Hellman is proposed before sending data to the cloud.

The analysis result shows that the suggested algorithm is quicker than the AES and Diffie–Hellman algorithms. The proposed algorithm is highly secure for cloud computing because it has the features of both algorithms [11]. Existing encipherment and decipherment algorithms (AES, Blowfish, DES and 3DES) are tested on the basis of different data settings. The simulation consequences display that the Blowfish algorithm achieves the best performance among all algorithms compared and that public and secret keys may be combined to obtain the best solutions for encryption. This approach aims to take advantage of the safety benefits of public key systems and the speed features of private key systems [12].

This approach is also used to remove the keys generated by the Diffie–Hellman algorithm for protection against eavesdropping (man-in-the-middle attack) and to generate randomly selected values under the following conditions:

1. A large value to realize computational infeasibility for attacker
2. Sufficiently random numbers such as pseudorandom numbers.

The Diffie–Hellman protocol has been applied to many security protocols including SSL, secure shell and IP security [13].

An overall performance evaluation of the most commonplace encipherment algorithms, in specifically, AES, RC2, RC6, DES, 3DES and Blowfish is also performed.

A laptop Pentium IV with a CPU 2.4 GHz is used in the tests. The laptop encrypts various file sizes, ranging from 321 KB to 7.139 MB. The following performance metrics are gathered:

1- Encipherment time

2- CPU process time

3- CPU clock cycles and battery power

The simulation effects display several points. The Blowfish algorithm achieves the first-rate performance amongst all selected algorithms in terms of changing packet size. In terms of converting information kind such text-to-image conversion, RC2, RC6 and Blowfish are time consuming, whereas 3DES has lower performance than DES does. Finally, in terms of changing key size, if the key length is big, the battery and time consumption obviously alternate [14].

A comparative evaluation among DES and Blowfish is carried out using distinctive parameters such as data type, data length and key length. The block cipher mode used is ECB.

A device equipped with Intel core i3-3120M (2.50 GHz) processor with Intel Q65 Express 4 GB of DDR3 RAM clocked of 1333 MHz and Microsoft Windows 8 is used to study the following instances:

Case 1: File with different data types. The results indicate that the Encipherment time does no longer different with the information type. Encipherment relies handiest on the quantity of bytes inside the file and not at the type of file. DES with a key size of fifty-six is faster than Blowfish.

Case 2: Data files of the same type with various sizes. Encipherment time increases as file size will increase in multiples of data length.

Case 3: Encryption algorithm with various key sizes. The outcomes display that DES-56 is quicker than Blowfish.

Case 4: Throughput. Throughput = plaintext (MB) / encryption or decryption time (in seconds). A device with Intel Core i3-3120M CPU with 2.50 GHz CPU speed and 4 GB RAM is used. The research results show that if the throughput worth will increase, then the power intake of this Encipherment method is reduced [15].

The modification of the S-DES algorithm is proposed to secure data by the use of pseudorandom key generation to create 16 bits of key. After creating 16 bits of key, the key is divided into 2 halves of 8 bits every that would include Key 1 and Key 2. The literature evaluate shows that many new researches have been performed in the previous few years. Considerable modifications and adjustments in the S-DES algorithm have also been made. In [4] improved the safety of the S-DES algorithm by means of including a few substitution strategies in S-DES.

DES is the most widely used encryption trendy algorithm. It employs 4 weak keys: 01010101 01010101, FEFEFEFE, E0E0E0E0 F1F1F1F1 and 1F1F1F1F 0E0E0E0E; consequently, weak keys may be prevented at key generation [16].

An algorithm that combines the manner of blending up bits and substituting boxes are proposed. High avalanche results are cited. A fantastic algorithm has high avalanche effects and it takes advantage of classical cryptography such as Playfair cipher, Vigenere cipher, Caesar cipher and embeds it with modern cryptography algorithms which include DES and Blowfish.

The technique has the subsequent steps:

The key has 64 bits or extra.

First of all, the message to be encrypted is break up into blocks of 64 bits. Then, each block is encrypted using the Playfair cipher. Then, the encrypted textual content undergoes an extensive blend up. Later, the blended textual content which additionally has 8 bits is further enciphered the usage of the Vigenere cipher. After that, the Vigenere ciphered textual content is break up into two components of 4 bits each. Those components are used to pick a value within the 16×16 substitution box. Then, the first 4 bits are considered as the row and the last 4 bits are considered as the column. Then, the 64 bits are XOR scrambled (M) instances (M=1, 2 or 3). After that, the 64 bits are cut up into 4 blocks of 16 bits each; those blocks are then XOR operated. Later, the blocks are similarly merged and XOR operated again. The entire process is performed N times, where N is between 1 and 16. Finally, the output is blended up the usage of S-box.

The avalanche effect is calculated using the following syntax:

Avalanche Effect = (Number. of flipped bits inside the ciphered text) / (Number. of bits within the ciphered text) × 100% consequently, the suggested algorithm has better avalanche impact than different existing algorithms (Playfair cipher, Vigenere cipher, Caesar cipher, DES and Blowfish) do.

The avalanche outcomes after undertaking the test are as follows:

Playfair cipher, 6.25%; Vigenere cipher, 3.13%; Caesar cipher, 1.25%; DES, 54.68%; Blowfish, 28.71%; proposed algorithm, 70.31% [17].

Security is more desirable with the aid of applying cryptographic techniques to avoid the leakage of questions for information manipulation consisting of marks. The proposed system has three parts essential to automate the whole paper-based totally grading technique.

First, a set of random questions is generated. Those questions are saved inside database, where the system selects exam's questions.

The system generates random questions and performs a secured distribution of questionnaires on the day of the exam, thereby stopping leakage or transmission of data from staff to students.

The faculty and administrators are furnished with a password. This is encrypted and decrypted via a cryptographic method.

Second, when the students take the exam, the answer sheets are scanned with an excessive-velocity scanner and evaluated on computer systems the usage of a mouse.

The server system needs for Linux, Apache, MySQL and PHP structure. The correction is made and the result is posted by mail. Finally, a student’s overall performance in the laboratory checks, on-line-COMPILER is evaluated to inspect the outcome for programs in laboratory assessments. As a result, the entire exam technique is computerized. This system guarantees a relaxed technique to carrying out exams and decreases the time required to assess and publish results [18].

A system for the Medium university diploma (AL-SHAMEL) exam is proposed. The AL-SHAMEL examination is the second maximum essential exam in Jordan.

On this proposed system, data encryption is utilized by one of the maximum vital cryptographic algorithms. This is an AES algorithm that facilitates the system (AL-SHAMEL) to method browser protection, safety of data, authorization, authentication and non-repudiation [19].

A comparative evaluation among AES and RC4 is performed the usage of a laptop with Intel CPU clocked at 2.99 GHz and 2 GB Random Access Memory.

The overall performance metrics (parameters) used in this research are throughput, memory utilization, CPU procedure time, encryption decryption time and key size variant.

The simulation effects are as follows:

a) Encryption time based on various packet sizes RC4 takes less time to encrypt a file than the AES algorithm does.
b) Decryption time based on various packet sizes RC4 takes less time to decipherment a file than the AES algorithm does.
c) Encryption time based on different key sizes. The three key length are 128-bit, 192 bit, and 256-bit. The results consequences show that the encipherment time for RC4 is nearly fixed and is much less than that of AES. Therefore, it consumes much less strength than AES does.
d) Decryption time based on various packet sizes RC4 is higher than AES.
e) Throughput for AES with various key sizes. The results show the superiority of RC4 to AES. For various modes of AES, throughput decreases as key length will increase; RC4 is speedy in nature and consumes little power.
f) Memory usage for AES and RC4 with distinct file sizes. AES consumes more memory than RC4 does.
g) Impact of converting packet length for encryption algorithms on CPU time. RC4 requires for extra time to encrypt a small-sized file, whereas AES requires for extra time to encrypt huge-sized files; for that reason, RC4 is useful for encrypting big data [20].

A comparative evaluation of the 3 algorithms namely DES, AES and RSA is accomplished.

Sure factors, together with computation time, memory usage and output byte are used to analyze the system.

A comparative evaluation is achieved for the encipherment algorithms given various sizes of data blocks and encipherment/decipherment speed; the same textual file is used for five tests.

The DES algorithm consumes the least encipherment time and the AES algorithm has the least memory utilization, whereas the encipherment time variance in the cases of AES and DES is minimal.

RSA consumes the longest encipherment time and its memory utilization is extraordinarily high; nevertheless, its output byte is the lowest [21].

A comparison of the most popular encipherment algorithms namely DES, Blowfish, CAST-128, RC6 and IDEA is supplied.

The assessment is accomplished on the premise of running time and throughput.

Various parameters, which include wide variety of rounds, file length, key size and key generation time, are used.

These encryption algorithms are applied in Java using the IAIK-JCE library in NetBeans IDE 7.0.1. Overall fulfillment is measured on a machine equipped with Intel(R) Core^{TM} i3 CPU M 370 @ 2.40 GHz and a 2.39 GHz 32-bit system with 4 GB of RAM and using on Windows 7.

To enhance the accuracy of the timing measurements, this system is completed ten times for each enter file.

The effects are summarized as follows:

a) RC6 is quicker than Blowfish which is quicker than CAST-128, which is faster than IDEA and DES.
b) Blowfish performs higher than IDEA does. IDEA has higher throughput for decryption than DES does, however the latter is advanced to the previous inside the context of encryption. The throughput of CAST-128 is favorite than that of IDEA, however the variation is minimal.

Therefore, Blowfish is suitable for packages in which keys do not change frequently and the usage of RC6 is useful when a high encryption rate is required [22].

The security of the S-DES algorithm is advanced via the transposition and shift row approach. In this manner, the original S-DES algorithm may be used for cryptography.

The steps of the enhanced S-DES algorithm are as follows:

a) Encryption with enhanced S-DES algorithm
   Plaintext → column transposition with multiple rounds to encryption → shift row → original S-DES algorithm
b) decryption with enhanced S-DES algorithm
   Ciphertext → original S-DES (for decryption) → inverse shift rows → column transposition with multiple rounds to encrypt → plaintext

The designed system improves the safety power of the original S-DES, although it requires additional computation; furthermore, cracking and breaking the enhanced S-DES algorithm is almost not possible [23].

6 of the most popular encipherment algorithms namely AES, DES, 3DES, Blowfish, RC2 and RC6 are evaluated.

A laptop equipped with IV 2.4 GHz CPU is used to perform the assessment.

The laptop encrypts a various file size that ranges from 321 KB to 7.139 and 139 MB for the text data, from 33 KB to 8262 KB for audio data and from 4006 KB to 5073 KB for video files.

The evaluation relies upon on some of parameters, such as size of data blocks, information type, battery power consumption, key length and encipherment /decipherment speed.

The test outcomes are as follows:

a) Blowfish is superior to other algorithms in terms of processing time.
b) RC6 requires the least amount of time amongst all algorithms, exclude Blowfish.
c) AES has a feature over 3DES, DES and RC2 in terms of time consumption and throughput.
d) 3DES has lower power consumption and throughput than DES does.
e) RC2 has lower overall performance and throughput than the alternative 5 algorithms do regardless of the small key length used [14].

The differential between DES, 3DES, and AES in terms of safety and overall performance is offered using nine elements, specifically, key size, encipherment kind, block length, improvement, cryptanalysis resistance, protection, potential key, possible ACS and printable character keys. The time need to test all viable keys is 50 billion seconds, thereby confirming that AES is best than DES and 3DES are.

Moreover, DES is designed to work higher in hardware than in software program. 3DES takes 3 times as a lot CPU power as DES does and designing AES is quicker in software and works efficaciously in hardware [24].

The SDES algorithm is better with the substitution cipher technique.

The improved cryptosystem has the following steps:

Plaintext → 4-square cipher technique → Trifid cipher substitution technique → SEDS algorithm → ciphertext

The advantages of this new technique are as follows.

The security is increased, brute force attacks are weak against the improved cryptosystem and brute force attacks require more time to hack the improved cryptosystem [3].

The performance of the AES and DES is evaluated in regarding of time of processing, CPU utilization, and encipherment throughput on Windows and Mac systems and given various text sizes.

The test is done on 2 devices: a laptop with Intel Core^{TM} i5 @ 2.5 GHz CPU running on Windows 7 and an Apple MacBook with Intel CPU Core i5 with Mac OS.

Visual Basic.NET 2013 is used to perform the test.

The simulation results are as follows.

In comparison with DES, AES is faster in terms of execution time on the 2 systems and has a higher throughput; moreover, DES requires less CPU usage than AES for the 2 systems [25].

Diverse cryptographic algorithms, example AES, DES and Blowfish, are in comparison on the premise of processing time and throughput. The experiments are evaluated on various video files.

The simulation end result shows the perfection of AES to other competitor regarding of throughput and processing time [26].

3 algorithms specifically DES, 3DES and RSA are analyzed on the basis of parameters which includes time taken to encrypt data and memory usage. The throughput is calculated via dividing the entire plaintext encrypted by total encipherment time for each algorithm. Java and ASP.NET are used for simulation. The results show that the speed of DES encryption is 2 times that of RSA encryption and that DES consumes less power than RSA does. 3DES nonetheless requires greater time than DES does and has more power consumption and less throughputs. The DES algorithm is advanced to the other algorithms in terms of power consumption and throughput. Moreover, the confidentiality and scalability provided by 3DES is a much better than that provided by RSA and DES, thereby making it an appropriate and secure algorithm [27].

A system that uses a combination of AES and Blowfish is proposed to eliminate the safety challenges of cloud storage. The Blowfish algorithm is implemented to the login web page to secure passwords and person names. When a file is uploaded and stored at the cloud, the Blowfish algorithm is carried out at the first level and then the AES algorithm is implemented at the second level to encrypt data. When downloading the file, the AES algorithm is implemented at the first level, after which the Blowfish algorithm is applied at the second one level to decrypt information. The multilevel encryption offers a high degree of safety [28].

Various algorithms are as compared on the premise of various factors which include their key length, block length, encipherment and decipherment key, scalability, algorithm, encipherment, decipherment, protection, power consumption, deposit of keys, inherent vulnerabilities, key used and rounds.

The comparison among DES, 3DES, RSA, AES, BLOWFISH and ECC is presented below.

Algorithm	Scalability
RSA	Not Scalable
DES	Scalable
AES	Not Scalable
3DES	Not Scalable
BLOWFISH	Scalable
ECC	Scalable

Algorithm	Encryption and Decryption Speed
RSA	High
DES	Low
AES	Low
3DES	Low
BLOWFISH	Low
ECC	Low

Algorithm	Security
RSA	Least Secure
DES	Not Secured Enough
AES	Excellent Secured
3DES	Excellent Secured
BLOWFISH	Least Secured
ECC	Average Secured

Algorithm	Inherent Vulnerabilities
RSA	Forced and Oracle attack
DES	Brute Forced, Linear and differential Cryptanalysis attack
AES	Brute Force Attack
3DES	Meet-in-the-middle-attack
BLOWFISH	Birthday Attack
ECC	Brute Force Attack

The 3DES algorithm is best for data protection because it uses 3 keys to encrypt and decrypt data [29].

The evaluation and assessment of a few symmetric key algorithms (RC4, AES, Blowfish, RC2, DES, Skipjack and 3DES) are provided on the basis of the following parameters: data type, data size, data density and key length. The divergence of encipherment time for various chosen cipher algorithms is likewise analyzed.

The running outcomes are obtained from a machine with Intel Core™ i7-2600 (3.40 GHz) processor with Intel Q65 Express four GB 1333 MHz DDR3 and Ubuntu 12.04 LTS operating device.

The Java platform (openjdk1.6.0_14) is used for implementation.

The outcomes are as follows:

Case Study 1: Files with various data types

The encryption time does not vary data type. Encryption relies most effective on the variety of bytes inside the file and not at the type of file. RC4 with a key length of 40 is the quickest among of the cipher algorithms examined.

Case Study 2: Data files of the same type with various sizes

Encipherment time increases as file size will increase in multiples of data size.

Case Study 3: Files with various data densities

The encipherment rate for a sparse and dense file is calculated. Outcomes display that encryption time is unaffected through the density of data in a file.

The encipherment rate for a specific cipher algorithm stays the identical even if the file is sparse or dense; it relies upon only on the number of bytes in the file.

Case Study 4: Encryption algorithms with various key sizes

The execution outcomes display that encryption time will increase with increasing key size for block ciphers and that RC4 is the fastest among all algorithms examined.

Moreover, encipherment time does not rely upon data type and the data density of the file. Encipherment only relies at the quantity of bytes within the file.

Moreover, the increase in key size increases encipherment time; the opposite is true for stream cipher such as RC4.

The AES algorithm is the fastest block cipher with an encipherment rate of 108 MB/s at bare minimum factor; however, the RC4 flow cipher with an encipherment rate of 270 MB/s emerges as the quickest among all analyzed cipher algorithms [30].

A multi cloud system is proposed to save customer records.

To permit protection in a multi cloud architecture, splitting and information encryption are incorporated. The data are split into 3 various cloud servers and saved in the multi cloud system. The overall performance evaluation of symmetric and asymmetric encipherment algorithms, consisting of DES, 3DES, Blowfish, AES, RSA and Diffie–Hellmen is based totally on factors or parameters inclusive of throughput, keys used, key size, computational speed, tunability, encipherment ratio and the security of information against attacks. The key size is high in the asymmetric encryption algorithms, so that complex codes in RSA can be broken. Among the symmetric key encipherment methods, the Blowfish algorithm is specified as the best solution. As for the asymmetric encipherment method, the RSA algorithm is the most secure because it makes uses the factoring of high prime numbers for key creation; consequently, the RSA algorithm can be applied to data safety in a multi cloud environment [31].

A secure scheme is proposed to guard the exam characteristics in numerous levels which include notations used, exam initialization, examination description, exam grades and examination revision.

The scheme relies upon on numerous encipherment protocols that provide strong protection level for all examination levels.

The public key infrastructure gives flexibility and scalability to the e-learning system and is identified as an enough device that offers privacy, validity, integrity and undeniable.

The lecturer encrypts an e-exam questions by the use of a relied on and licensed public key.

The trusted and authorized private key is needed to obtain the exam questions and this key is confined to rely on authorities.

The lecturer can then send the exam answers to trusted authorities. Thus, the answer is encrypted and can only be obtained by the latter.

The authorized user encrypts the e-examination solution using the lecturer’s public key.

The students’ answers are stored secret and only the lecturer and the trusted authority have access to them [32].

The encryption algorithm of an online examination system (OES) is studied, and the algorithms 3DES, AES, RSA and Blowfish are compared with DES.

A new algorithm extra secure-DES (XS-DES), which is more secure than DES and faster than 3DES is proposed.

The proposed algorithm uses a 128-bit key instead of a 64 bit and it splits the key into 2, the left and right parts, each of which comprises 64 bits.

The framework is illustrated as follows.

When a student enters the Online Examination System, he has 2 options namely to login and to register.

Login includes the Admin, Examiner and Student logins where the Admin can add or remove examinations or examiners.

The examiner can assess the exam and send the result; however, he cannot see the students’ name or identification because the system encodes students’ names by applying the suggested method.

To obtain the results, the Online Examination System sends an email containing the decrypted results. The system initially validates the student’s authority to view the decrypted results through the answer to a security question, which has been determined before the exam is taken [33].

The encryption algorithms consume a considerable amount of computing resources such as memory and computation time. A comparative analysis between RSA and AES algorithm based on some parameters such as encryption time and memory usage, is conducted. The performance evaluation depends on different file formats such as text files, PDF files, Microsoft word files and images. RSA has greater memory usage than AES does, but it requires less encryption time [34].

The literature review is summarized for comparison based on factors such as encipherment and decipherment time and throughput. The summary is shown in Table 1.

Table 1: Comparison of various cryptographic algorithm

References	Evaluation Parameters	Compared Algorithms	Findings	Domain
[7]	1- Security	1- RSA 2- Diffie-Hellman 3- Both RSA and Diffie-Hellman	Diffie Hellman Key can be used with RSA, DES, AES and Elliptical Curve Cryptography. Combining both RSA Diffie-Helman can perform stronger than RSA or Diffie-Hellman.	Text file
[8]	1- Security 2- Encryption time	1- DES 2- 3DES 3- AES 4- Blowfish	Using 3DES with ECB and CBC has better performance	Simulation in .Net Classes
[11]	1- Encryption time 2- Security	1- AES 2- Diffie Hellman	The proposed algorithm (combining both AES and Diffie Hellman) perform better in terms of encryption time and security.	Cloud Computing

[12]	1- Encryption time 2- Decryption time	1- DES 2- 3DES 3- AES 4- BLOWFISH	Blowfish has better performance than other competitors	Text Files
[14]	1- Throughput 2- Encryption time 3- Decryption time 4- Power consumption	1- DES 2- 3DES 3- AES 4- BLOWFISH 5- RC6 6- RC2	Blowfish has better performance than other competitors	Audio files Video files Text files
[15]	1- Encryption time	1- DES 2- BLOWFISH	The encryption time will increase by increasing the key size, but in DES the key size has no effect on encryption time	XML files Video files
[16]	1- Encryption time	1- DES 2- Modified DES	Modified DES performed better in terms of encryption time	Text files Images
[17]	1- Security	1- DES 2- BLOWFISH 3- Playfair cipher 4- Vigenere cipher 5- Caesar cipher 6- Proposed algorithm	The proposed algorithm showed better results in terms of security since it has better avalanche	Text files
[20]	1- Throughput 2- Memory utilization 3- Encryption time 4- Decryption time	1- AES 2- RC4	RC4 better than AES in terms of all evaluated parameters	Text file
[21]	1- Computation time 2- Memory Utilization	1- DES 2- AES 3- RSA	DES has better encryption time, while AES has memory usage. on the other hand, RSA algorithm produce small-size output file	Text file
[22]	1- Encryption time	1- DES 2- BLOWFISH 3- RC6 4- IDEA 5- CAST-128	RC6 the fastest encryption algorithm. Blowfish better for applications that does not require changing key.	Text file
[23]	1- Security 2- Computation time	1- S-DES 2- Enhanced S-DES	Enhanced S-DES more secure than S-DES, while it takes longer time for encryption	Text file
[24]	1- Security	1- DES 2- 3DES 3- AES	AES the most secure algorithm among all competitors	Text file
[3]	1- Security	1- S-DES 2- Enhanced S-DES	Enhanced S-DES more secure than S-DES, but needs more time for encryption	Text file
[25]	1- Processing Time 2- CPU Usage 3- Throughput	1- DES 2- AES	AES is faster and give high throughput DES consume less CPU usage	Text file
[27]	1- Encryption Time 2- Power consumption 3- Throughput	1- DES 2- 3DES 3- RSA	3DES more secure. DES consume less power, memory and encryption / decryption time	Text file
[29]	1- Encryption time 2- Decryption time 3- Security 4- Power consumption	1- DES 2- 3DES 3- AES 4- BLOWFISH 5- RSA 6- ECC	3DES and AES more secure. AES, Blowfish, ECC are faster encryption / decryption time. RSA has higher power consumption over all.	Text file

[31]	1- Throughput 2- Encryption ratio	1- DES 2- 3DES 3- AES 4- BLOWFISH 5- Diffie Hellman	RSA more secure in cloud environment.	Cloud environment

As it can be seen from Fig. 8 that most of the researcher focused their research on DES, 3DES, Blowfish and AES algorithms and mainly in the field of encryption and decryption time while RC6, RC4, RC2 ECC and D-H have the lowest research interest among all algorithms.

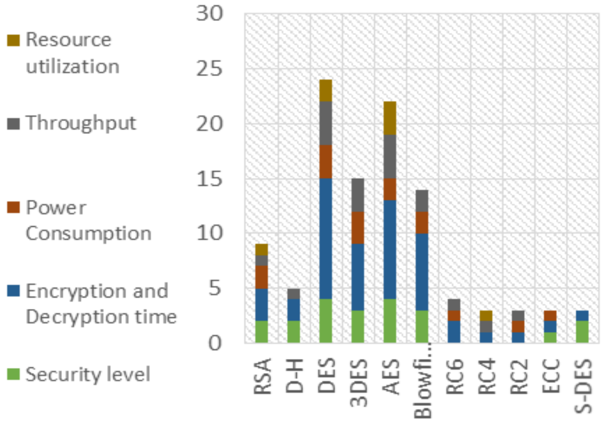

Fig. 8: Number of researches vs tested parameters

IV. CONCLUSION

The cryptographic algorithms vary in terms of parameters which includes encipherment and decipherment time, memory, throughput and CPU utilization.

This research analyzes the want to improve a combine encipherment algorithm that mixes various encipherment algorithms on the basis of all appropriate factors that are used to increase the overall safety and security of encipherment methods.

V. ACKNOWLEDGMENT

The authors thank the Government of Malaysia for funding this research under the Fundamental Research Grant Scheme (FRGS/1/2018/ICT03/UNISZA/02/2) and also Universiti Sultan Zainal Abidin, Terengganu, Malaysia.

REFERENCES

1. C. P. Pfleeger, S. L. Pfleeger, and J. Margulies, Security in Computing. New Jersey: Prentice Hall, 2015.
2. M. Mushtaq Faheem, S. Jamel, A. Hassan Disina, Z. A. Pindar, N. Shafinaz Ahmad Shakir, and M. Mat Deris, "A survey on the cryptographic encryption algorithms," Int. J. Adv. Comput. Sci. Appl., 8(11), 2017, pp. 333-344.
3. G. Suman, C. Krishna, and M. T. Se, "Improved cryptosystem using SDES algorithm with substitution ciphers," International Journal of Advanced Research in Computer Science and Software Engineering, 3(7), 2013, pp. 131-136.
4. S. N. Habib, R. Awan, and W. Haider, "A modified simplified data encryption standard algorithm," International Journal of Computer Science and Software Engineering, 6(7), pp. 152-154, 2017.
5. W. Stallings, and M. P. Tahiliani, Cryptography and Network Security: Principles and Practice. London: Pearson, 2014.
6. D. Costinela-Luminita, "Information security in e-learning platforms," Procedia - Soc. Behav. Sci., 15, 2011, pp. 2689-2693.
7. R. Kumar, and C. C. Ravindranath, "Analysis of Diffie Hellman Key Exchange Algorithm with proposed Key Exchange Algorithm," Int. J. Emerg. Trends Technol. Comput. Sci., 4(1), 2015, pp. 40-43.
8. S. Karthik, and A. Muruganandam, "Data Encryption and Decryption by using Triple DES and performance analysis of crypto system," Int. J. Sci. Eng. Res., 2(11), pp. 24-31, 2014.
9. V. Agrawal, S. Agrawal, and R. Deshmukh, "Analysis and review of encryption and decryption for secure communication," Int. J. Sci. Eng. Res., 2(2), 2014, pp. 2347-3878.
10. V. Sukhraliya, S. Chaudhary, and S. Solanki, "Encryption and Decryption Algorithm using ASCII values with substitution array approach," International Journal of Advanced Research in Computer and Communication Engineering, 2(8), 2013, pp. 3094-3097.
11. R. Malik, and P. Kumar, "Cloud computing security improvement using Diffie Hellman and AES," Int. J. Comput. Appl., 118(1), 2015, pp. 975-8887.
12. V. K. Mitali, and A. Sharma, "A survey on various cryptography techniques," Int. J. Emerg. Trends Technol. Comput. Sci., 3(4), 2014, pp. 307-312.
13. M. Ahmed, B. Sanjabi, D. Aldiaz, A. Rezaei, and H. Omotunde, "Diffie-Hellman and its application in security protocols," Int. J. Eng. Sci. Innov. Technol., 1(2), 2012, pp. 69-73.
14. D. S. Abd Elminaam, H. M. A. Kader, and M. M. Hadhoud, "Evaluating the performance of symmetric encryption algorithms," Int. J. Netw. Secur., 10(3), 2010, pp. 213-219.
15. B. L. Srinivas, A. Shanbhag, and A. S. D. Souza, "A comparative performance analysis of DES and BLOWFISH symmetric algorithm," International Journal of Innovative Research in Computer and Communication Engineering, 2(5), 2014, pp. 77-88.
16. M. A. Hameed, A. I. Jaber, J. M. Alobaidy, and A. Alaa, "Design and simulation DES algorithm of encryption for information security," American Journal of Engineering Research, 7(4), 2018, pp. 13-22.
17. S. Ramanujam, and M. Karuppiah, "Designing an algorithm with high avalanche effect," Int. J. Comput. Sci. Netw. Secur., 11(1), 2011, pp. 106-111.
18. R. Divya, and M. Kumar, "Enhanced digital assessment of examination with secured access," International Journal of Advanced Studies in Computers, Science and Engineering, 3(10), 2014, pp. 33-37.
19. M. M. Al-Laham, "Reducing security concerns when using cloud computing in online exams case study: General Associate Degree Examination (Shamel) in Jordan," Int. J. Comput. Sci. Inf. Technol., 7(6), 2015, pp. 131-144.
20. N. Singhal, and J. P. S. Raina, "Comparative analysis of AES and RC4 algorithms for better utilization," Int. J. Comput. Trends Technol., 2(6), 2011, pp. 177-181.

21. S. M. Seth, and R. Mishra, "Comparative analysis of encryption algorithms for data communication 1," International Journal of Computer Science and Technology, 2(2), 2011, pp. 292-294.
22. K. Aggarwal, "Performance evaluation of RC6, Blowfish, DES, IDEA, CAST-128 Block Ciphers," Int. J. Comput. Appl., 68(25), 2013, pp. 10-16.
23. A. Akhtar, M. Zia, and U. Baig, "Enhancing the security of simplified DES algorithm using transposition and shift rows," International Journal of Computer Science and Software Engineering, 6(5), 2017, pp. 115-119.
24. H. Alanazi, B. B. Zaidan, A. A. Zaidan, H. A. Jalab, M. Shabbir, and Y. Al-Nabhani, "New comparative study between DES, 3DES and AES within nine factors," Journal of Computing, 2(3), 2010, pp. 152-157.
25. S. D. Rihan, S. E. F. Osman, and A. Khalid, "A performance comparison of encryption algorithms AES and DES," Intetnational J. Eng. Res. Technol., 4(12), 2015, pp. 151-154.
26. S. Pavithra, "Performance evaluation of symmetric algorithms," J. Glob. Res. Comput. Sci., 3(8), 2012, pp. 43-45.
27. M. Marwaha, R. Bedi, A. Singh, and T. Singh, "Comparative analysis of cryptographic algorithms," Int. J. Adv. Engg. Tech., 4(3), 2013, pp. 16-18.
28. U. Gupta, S. Saluja, and T. Tiwari, "Enhancement of cloud security and removal of anti-patterns using multilevel encryption algorithms," Int. J. Recent Res. Asp., 5(1), 2018, pp. 55-61.
29. M. Mohan, and J. Prakash, "Analysis of various cryptographic algorithms," International Journal of Engineering Technology, Management and Applied Sciences, 2(3), 201, pp. 51-61.
30. S. Singh and S. Kumar, "Analysis of various cryptographic algorithms," Int. J. Advanced Research in Science, Engineering and Technology, 5(3), pp. 5341-5348, 2018.
31. M. T. Bemila, K. Kundar, L. Jain, S. Sharma, and N. Makasare, "Comparative study of various security algorithms applicable in multi-cloud environment," Int. J. Advanced Research in Computer and Communication Engineering, 5(3), 2016, pp. 460-463.
32. S. J. Aboud, "Secure e-exam scheme," International Journal of Science and Research, 3(9), 2014, pp. 2200-2203.
33. O. Zughoul, H. M. Jani, A. Shuib, and O. Almasri, "Privacy and security in online examination systems," IOSR J. Comput. Eng., 10(4), 2013, pp. 63-70.
34. K. P. Karule and N. V Nagrale, "Comparative analysis of encryption algorithms for various types of data files for data security," International Journal of Scientific Engineering and Applied Science, 2(2), 2016, pp. 495-498.