Evaluation of Four Encryption Algorithms for Viability, Reliability and Performance Estimation

J. B. Awotunde[1*], A. O. Ameen[1], I. D. Oladipo[1], A. R. Tomori[2], M. Abdulraheem[2]

[1]Department of Computer Science, University of Ilorin, Ilorin, Nigeria.
[2]Computer Sciences and Information Technology, University of Ilorin, Ilorin, Nigeria.

ABSTRACT: Data and information in storage, in transit or during processing are found in various computers and computing devices with wide range of hardware specifications. Cryptography is the knowledge of using codes to encrypt and decrypt data. It enables one to store sensitive information or transmit it across computer in a more secured ways so that it cannot be read by anyone except the intended receiver. Cryptography also allows secure storage of sensitive data on any computer. Cryptography as an approach to computer security comes at a cost in terms of resource utilization such as time, memory and CPU usability time which in some cases may not be in abundance to achieve the set out objective of protecting data. This work looked into the memory construction rate, different key size, CPU utilization time period and encryption speed of the four algorithms to determine the amount of computer resource that is expended and how long it takes each algorithm to complete its task. Results shows that key length of the cryptographic algorithm is proportional to the resource utilization in most cases as found out in the key length of Blowfish, AES, 3DES and DES algorithms respectively. Further research can be carried out in order to determine the power utilization of each of these algorithms.

KEYWORDS: Algorithm, Cryptography, Encryption, Evaluation, Performance, Decryption

[Received May 19 2016; Revised December 13 2016; Accepted December 20 2016]

I. INTRODUCTION

The use of information technology in our everyday human endeavour is no longer a statement, its usage has primary means of driving and running every aspect of everyday lives as covered areas like education, banking, governance, business, military, power, health and so on. These have becomes part of human lives, since someone cannot do without the use of information technologies on a daily basis. Hence, as the use of Information Technology increases the value of information it spreads as also increases and with the enforcement of cashless society in developing world, which means even monetary transactions are now channelled through IT thereby transferring the risk of carrying cash to the use of the IT, which add to the burden of secured data is being applied for the cash transfer.

Securing data or information is when tactics measures are taken to ensure that the information or data is not tampered with and the integrity of the information is assured. One of the most common approaches is data encryption (cyphering) which involves using methods to make readable information become unreadable (Ankita, et al, 2016).

With the introduction of computer, the need for automated tools for protecting files and other information stored on the computer become evident. This is especially the case for a shared system, such as a time sharing-system and the need are even more acute for systems that can be accessed over a public telephone network, or the internet. As the amount of network-attacked data and storage systems grow, so does the exposure to data loss and intrusion increased. Also, the unauthorized access by theft has grown daily because of the unsecured data and information on the computer system.

II. CRYPTOGRAPHIC ALGORITHMS

Cryptography has roots that began around 2000 B.C. in Egypt when hieroglyphics were used to decorate tombs to tell the story of the life of the deceased. The practice was not as much to hide the messages themselves, but to make them seem more noble, ceremonial, and majestic (HarrisX, 2001). There are several ways of classifying cryptographic algorithms. In this paper, the four algorithms will be briefly discussed base on the number of keys that are employed for encryption and decryption.

Cryptography is an essential part of modern world information security making the virtual world a safer place (Priyadarshini, Prashant, Narayan, & Meena, 2016). Cryptography is a process of making information unintelligible to an unauthorized person (Priyadarshini et al, 2016). Cryptography also termed as "secret writing" is a science of concealing information so that only the intended parties can have access to the private information (Ankita, Paramita, & Sunita, 2016). It protects the privacy and modification of data which may occur due to active and passive attacks in the channel. Cryptographic methods such as symmetric and asymmetric encryption algorithm ensure integrity, privacy, nonrepudiation and validity of secret data (Ankita, et al, 2016).

*Corresponding author's e-mail address: jabonnetbylinks@gmail.com
doi: http://dx.doi.org/10.4314/njtd.v13i2.5

Shared key cryptography is also called symmetric algorithm (Bala & Kumar, 2015). During data transmission, the sender and the receiver share the same key for encryption and

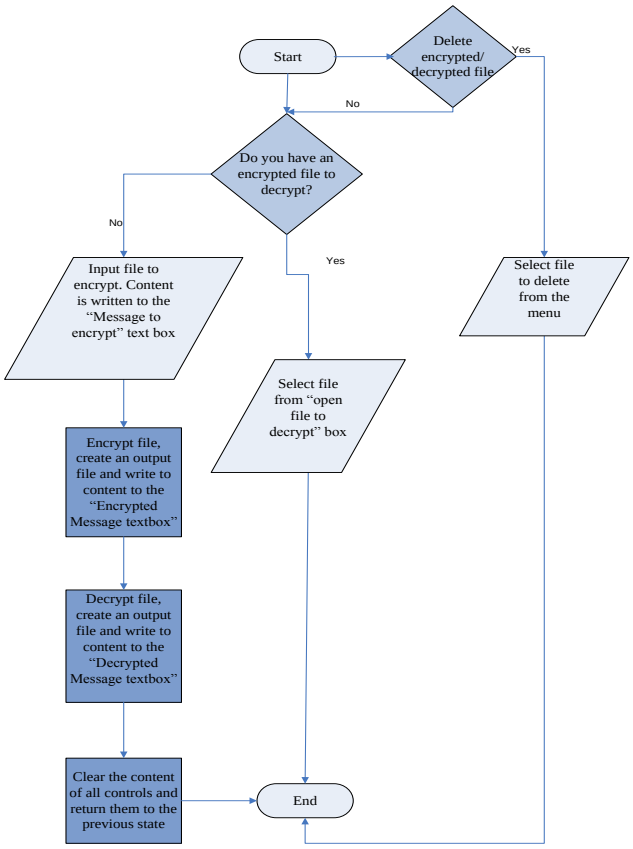

Figure 1: Flowchart for Encryption and Decryption (Singh and Maini, 2011) used.

decryption. To maintain confidentiality, this key needs to be kept secured. If the key for communication is leaked out the data can be stolen by the attacker (Bala & Kumar, 2015). Symmetric key cryptographic ciphers have different structures that are used to construct the block of the different Symmetric key block ciphers (Ranjeet, Vivek, & Jibi, 2015).

There are symmetric key structures like Feistel network, Substitution-permutation network etc. In the case of Fiestel network, the encryption and decryption process of the block are almost similar to each other, except it requires the reversal of key schedule (Ranjeet et al, 2015). Iteration is a characteristic feature of Fiestel network cipher as an internal function knows as round function. There are different types of symmetric algorithms like Data Encryption Standard (DES), Triple Data Encryption Standard (3DES), Advanced Encryption Standard (AES) and Blowfish (Bala & Kumar, 2015).

Asymmetric Algorithm is also called public key cryptography. It uses two keys 'Private key' and 'Public key'. During data transmission, the sender encrypts the plain text with the help of public key known as the cipher text and the receiver decrypts this cipher text with the help of its private key (Bala & Kumar, 2015). The different types of asymmetric algorithms are Rivest Shamir Adlemen (RSA), Diffie-Hellman and Digital Signature Algorithm.

This paper reviews four encryption algorithms by testing their memory consumption rate, different key size, CPU utilization time period and encryption speed utilized by each algorithm. The four algorithms were selected because of their stress-free implementation and the algorithms are the most common use in cryptosystems.

Brief background of the four algorithms in the Experiment

A. Data Encryption Standard (DES)

DES is a symmetric key block cipher. The key length is 56 bits and block size is 64 bit length. It is vulnerable to key attack when a weak key is used. DES was found in 1972 by IBM using the data encryption algorithm. It was adopted by the government of USA as standard encryption algorithm. It began with a 64 bit key and then the NSA put a restriction to use of DES with a 56- bit key length, hence DES discards 8 bits of the 64 bit key and then uses the compressed 56 bit key derived from 64 bit key to encrypt data in block size of 64-bits .DES can operate in different modes - CBC, ECB, CFB and OFB, making it flexible. It is vulnerable to key attack when a weak key is used. In 1998 the supercomputer DES cracker, with the help of lakh's of distributed PCs on the Internet, cracked DES in 22h (Priyadarshini, 2016).

B. Triple Data Encryption (3DES)

In cryptography, 3DES is a block cipher. 3DES was first published in 1998 which gets its name so because it applies DES cipher three times to each block of data, Encryption – Decryption – Encryption using DES. The key length is 112 bits or 168 bits and block size is 64 bit length. Because of the increasing computational power available these days and weak of the original DES cipher, it was subject to brute force attacks and various cryptanalytic attacks; Triple DES was designed to provide a relatively simple method of increasing the key size of DES to protect against such attacks, without designing a completely new block cipher algorithm (Karthik & Muruganandam, 2014, Priyadarshini, 2016).

C. Advance Encryption Standard (AES)

AES algorithm was developed in 1998 by Joan Daemen and Vincent Rijmen, which is a symmetric key block cipher. AES algorithm, supports any combination of data and key length of 128, 192, and 256 bits. AES allows a 128 bit data length that can be split into four basic operational blocks. These blocks are considered as array of bytes and organized as a matrix of the order of 4×4 which is also called as state and subject to rounds where various transformations are done. For full encryption, the number of rounds used is variable N = 10, 12, 14 for key length of 128,192 and 256 respectively. Each round of AES uses permutation and substitution network, and is suitable for both hardware and software implementation (Ritupahal & Vikaskumar, 2013, Priyadarshini, 2016).

D. Blowfish

Blowfish was first published in 1993 .It is a symmetric key block cipher with key length variable from 32 to 448 bits and block size of 64 bits. Its structure is fiestal network. Blowfish is a symmetric block cipher that can be used as a informal replacement for DES or IDEA. It takes a variable-length key, from 32 bits to 448 bits, making it ideal for both domestic and commercial use. Blowfish was designed by Bruce Schneier as a fast, free alternative to existing encryption algorithms. From then it has been analyzed considerably, and it is slowly gaining popularity as a robust encryption algorithm. Blowfish is not patented, has free license and is freely available for all uses (Pratap, 2012), (Priyadarshini, 2016).

The paper is organized as follows. In the next section, related work were discussed stating the parameters used in evaluate the algorithms and the results obtained. In Section III, detailed of type of computing platform, CPU specification, range of file size for encrypted/decrypted data and description of parameters used for the algorithms are explained. A comparative analysis of the algorithms is explored in Section IV. Finally, the paper is concluded with the subsequent Section.

III. RELATED WORKS

Over the years lots of algorithms have been implemented looking at the weakness and strength of the existing approaches to secure data. These algorithms are classified under the symmetric and asymmetric algorithms. Some of the results obtained from other research papers give more insight about the performance of the compared algorithms such as Blowfish and AES algorithms. It was identified from (Chandramouli, 2006; Hiran, 2003) that AES operates faster and more efficient than other symmetric encryption algorithms. When the transmission of data is considered there is insignificant difference in performance of different symmetric key schemes (most of the resources are consumed for data transmission rather than computation).

A study in (Nadeem & Javed, 2005) is conducted for different popular secret key algorithms such as DES, AES, and Blowfish. They were implemented, and their performance was compared by encrypting input files of varying contents and sizes. The algorithms were tested on two different hardware platforms, to compare their performance. The algorithms were implemented on two different machines: P-II 266 MHz and P-4 2.4 GHz. The results showed that Blowfish had a very good performance compared to other algorithms. Also it showed that AES had a better performance than 3DES and DES.

In (idrus, Ahjunid, & Ais, 2008; Krishnamurthy et al, 2008) a study of security measure level has been proposed for a web programming language to analyze four Web browsers. This study measured the performances of encryption process at the programming script with the Web browsers. This study conducting tests simulation in order to obtain the best encryption algorithm versus Web browser. It was concluded in (idrus, Ahjunid, & Ais, 2008) that Rijndael and Twofish achieved the highest performance on Ultras ARC, Pentium II and Itanium. It was therefore recommended that Rijndael and Twofish should be used on these system since both outperformed AES algorithms.

Dhawan (2002) compared the performance of the different encryption algorithms by conducting experiments using .NET framework. The comparison was performed on the following algorithms: DES, 3DES, RC2, and AES (Rijndael). It was concluded that AES outperformed other algorithms in both the number of requests processes per second in different user loads, and in the response time in different user-load situations (Singh, Kumar, & Sandha, 2005; Agrawal & Mishra, 2011).

In (Seth & Mishra, 2011), AES and RSA considering certain parameters such as computation time, memory usages and output byte. A cryptographic tool was used for conducting experiments. It was concluded that RSA consumes longest encryption time and the memory usage is also very high but output byte is smallest in case of RSA algorithm. Based on the text files used and the experimental result it was concluded that DES consume least encryption time and AES has least memory usage while encryption time difference is very minor in case of AES algorithm and DES algorithm.

Apoorva, and Kumar (2013) compared most common symmetric cryptography algorithms: AES, TWOFISH, CAST-256 and BLOWFISH. The comparison was made on the basis of these parameters: speed, block size, and key size. It was concluded that blowfish is superior to other algorithm as it takes less time. Although when the data size was very small this difference was insignificant. But for file having size greater than 100 KB was very clearly significant.

Abdul et al (2009) discussed six most common encryption algorithms such as AES (Rijndael), DES, 3DES, RC2, BLOWFISH and RC6. These algorithms were compared and their performance were evaluated. A comparison were conducted using the following parameters: sizes of data blocks, different data types, battery power consumption, different key size and finally encryption speed. It was concluded that there is no significant difference when the results are displayed either in Hexadecimal Base encoding or in Base 64 encoding. Secondly in the case of changing packet size, it was concluded that BLOWFISH has better performance than other common encryption algorithms used, followed by RC6. Also, it was found that 3DES still has low performance compared to algorithm DES. Finally in the case of changing key size, it can be seen that higher key size leads to clear change in the battery and time consumption.

Marwaha et al (2013) compared and evaluated three algorithms DES, 3DES and RSA. DES and 3DES are symmetric key algorithms and RSA is an asymmetric key algorithm. Algorithms have been analyzed on their ability to secure data, time taken to encrypt data and throughput the algorithm requires. Performance of different algorithms was different according to the inputs. It was concluded that confidentiality and scalability provided by 3DES over DES and RSA is much higher and makes it suitable even though DES consumes less power memory and time to encrypt and decrypt the data, but on security DES can be easily broken by brute force technique as compared to 3DES and RSA, making it the last secure algorithm.

Alam and Khan (2013) discussed performance and efficiency analysis of different block cipher algorithms

namely: DES, 3DES, CAST- 128, BLOWFISH, IDEA and RC2 of symmetric key algorithm. The parameters used were: input size of data (in the form of text, audio and video), encryption time, throughput of encryption of each block cipher and power consumption. It was concluded that 3DES has more power consumption and less throughput than the DES due to its triple phase characteristics.

Hamdan et al (2010) has done the comparative analysis of three Encryption Algorithms (DES, 3DES and AES) within nine factors such as Key Length, Cipher Type, Block Size, Security, Possible Keys, Possible ASCII printable character keys and Time required to check all possible keys at 50 billion keys per second etc. Study shows that AES is better than DES and 3DES. Chadi & Pierre (2015) find in quantitative terms like Speed-Up Ratio that benefits of using cloud resources for implementing security algorithms (RSA, MD5 and AES) which are used by businesses to encrypt large volumes of data. Three different kinds of algorithms are used RSA (an asymmetric encryption algorithm), MD5 (a hashing algorithm) and AES (a symmetric encryption algorithm). The paper concludes that the algorithms implemented on cloud environment (i.e. Google App) are more efficient than using them on single system.

For both uni-processor (local) as well as cloud (Appengine) environment, RSA is the most time consuming and MD5 is the least. Highest Speed-Up Ratio is obtained in AES for low input file sizes and the Speed-Up Ratio falls sharply as the input file size is increased. For each input size, the Speed-Up Ratio is highest for AES, followed by MD5 and least for RSA algorithm. This study aims at finding the best out of AES, DES, 3DES and Blowfish algorithms used in evaluate the memory construction rate, different key size, CPU utilization time period and encryption speed of each algorithm. The results show the impact of changing data load on each algorithm and the impact of Cipher Mode (Encryption Mode) used on the algorithms.

IV. IMPLEMENTATION.

This study implemented the algorithms in java programing language using Eclipse IDE. The authors used packages java safety and java crypto .The packages java crypto and security provides security features like encryption, decryption, key generation, key management infrastructure, authentication and authorization features. This study implemented blowfish in java, converted into a jar and added blowfish jar to crypto library externally since blowfish is not provided in java security and crypto library. The files of size used were between 512KB to 1.5MB, which consisting of text and images as input for encryption. The encrypted output of each file is saved as a file, which in turn is input for decryption. For the sake of comparison the study used the same input files for all algorithms throughout the experiment. Also the study used the same system for all implementations and analysis work, so that memory and processor conditions remain same for all algorithms for comparison. All block cipher algorithms are set in a same mode ECB which is default in java crypto and security. Java crypto and security package contains the classes and interfaces that implement the Java security architecture.

These classes can be broadly divided into two categories. First, the classes that implements cryptography to perform operations for information to be transmitted. Second, there is authentication and access control classes that implement message digests and digital signatures and can authenticate users and other objects. Using the libraries of this package, we implement various cryptographic algorithms making minor changes in the calling functions. The method of implementing algorithms using functions of java.security and java.crypto package is as follows: Generate key using key generator class, create a cipher object with parameters algorithm name and mode, initialize the cipher created for encryption and perform encryption using doFinal() method.

A. System Parameters

The experiment are conducted using CORE i5 64bit processor with 4GB of RAM. The simulation program is compiled using the default settings in .NET 2013 visual studio for C# windows applications. The experiments will be performed couple of times to assure that the results are consistent and are valid to compare the different algorithms. The value of the information is proportional to the risk of information which means when high valued information there will be a great need for the information to be protected and secured.

Encryption algorithms consume a significant amount of computing resources such as CPU time, memory and computation time (Mandal, 2012). Determining the appropriate algorithm suited for a particular data type, and scenario there is need to answer the following questions and possibly problems;

1. To determine time taken by an algorithm to processes a file during encryption and decryption.
2. To estimate the amount of CPU time consumed in the process.
3. To Calculate the Memory Utilized.
4. How much success rate have been recorded in breaking the algorithms

This study is based on the design of a benchmark application using java program for testing the selected encryption algorithms (DES, 3DES, AES and BLOWFISH) in order to evaluate the resource utilization and time consumed by each algorithm with different input data size to determine the most appropriate algorithm for each data type and scenario. Various encryption algorithms have been developed in time past for various purposes. They all have their strengths for encryption and decryption but they also have their weaknesses in times of attack (i.e unauthorized person(s) attempting to decipher encrypted data in a forceful manner without the appropriate decryption keys). Encryption algorithms are said to have improved from time to time but not all of them are reliable for every kind of data to be ciphered.

That is why this study embarks on testing four of the most commonly used encryption algorithms to determine which one is best suitable and at what time. Due to the fact that four of these encryption algorithms will be implemented to achieve

the aim of the paper, it is important to note that the work does not primarily focus on the development of the encryption algorithms, but on the resource utilization of this algorithm to determine if they are suitable for the task at which they are implemented on.

Each of the four algorithms will be implemented solely for the testing of their encryption and decryption strengths, weaknesses as well as the resources they make use of during the process.

Each of the encryption techniques has its own strong and weak points. In order to apply a suitable cryptography algorithm to an application, we should have knowledge regarding performance, strength and weakness of the algorithms. Therefore, these algorithms must be analyzed based on several features. In this paper, analysis is done with following metrics under which the cryptosystems can be compared are described below:

B. Encryption Speed

It is important that the encryption and decryption algorithms are fast enough to meet real time requirements. Therefore, the speed at which encryption and decryption takes place in each of the case study algorithms will also be determined. This is carried out by logging the time of the start of the encryption process and login the finish time of the encryption/ decryption process.

Start time = Tx_1 (ms)

Finish time = Tx_2 (ms)

Therefore time taken to complete process is given in eqn (1).

$$Tx = Tx_2 - Tx_1 \quad (1)$$

C. Different Key Size

In the encryption methodologies the key management is the important aspect that shows how the data is encrypted. The image loss the encryption ratio is based on this key length. The symmetric algorithm uses a variable key length which is longer. Hence, the key management is a considerable aspect in encryption processing which will be studied amongst the four (4) algorithms. Each algorithm utilizes a specific number of key length which is used as a seed in the process block.

D. CPU Utilization Time Period

Each algorithm makes use of system resources such as memory, CPU, and so on. Therefore the amount of resources used by each algorithm will be logged and compared to determine which algorithm chomp extra size of system resources and how much system resource it used in performing it task.

E. Memory Consumption Rate

Different encryption techniques require different memory size for implementation. This memory requirement depends on the number of operations to be done by the algorithm, key size used, initialization vectors used and type of operations. The memory used impacts cost of the system. It is desirable that the memory required should be as small as possible.

V. SIMULATION PROCEDURE

This application is divided into three modules for simplicity in understanding of the application. The first module (data module) involves the specification of the data to be encrypted or decrypted using the four case study algorithms. The second module (algorithm module) involves the selection of any of the four implemented algorithm to be used for the encryption or decryption of the data specified in the data module. The third module (report module) involves the display of the resulting report of the encryption or decryption process based on all the performance factors specified above. This report will serve as data that will be used to analyse each of the four algorithms based on their performances. This module will also involve logging the reports generated to the appropriate database files.in the range 7.0 to 8.0 bit per character. The algorithm doesn't works well with the test data.

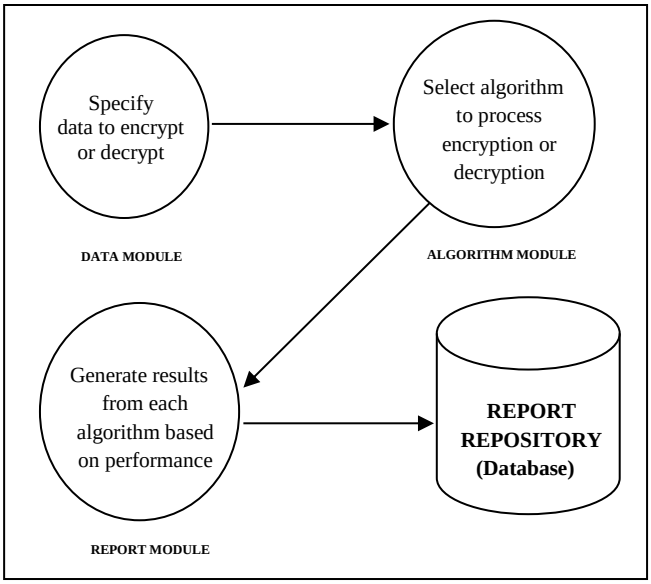

Figure 2: An outline of the Java Program Modules.

A. Data Module

As described above, the data module is the phase where the data to be encrypted or decrypted is entered. Data occurs in various types such as numeric data (with or without floating points e.g. 1234, 666889.453), alpha-numeric data (a combination of alphabets and numbers e.g. A45FBE3), alphabets or characters (e.g. this is my "project"). Data to be encrypted are images, text files, folders of important documents with different file extensions etc. in order to accommodate every data type to be encrypted, the data module consist of three platforms.

B. Algorithm Module

This module involves the process of selecting the encryption algorithm to be made used of. After entering the

data to be encrypted or decrypted in the data module, the user selects each of the case study algorithms to encrypt or decrypt the data. Each implemented algorithm goes through its process of encryption or decryption to produce a result. The data entered in the data module is encrypted or decrypted once with each algorithm (four times in all). The resource utilized, according to the performance evaluation factors, for each algorithm process for the specified data is being recorded to be displayed in the report module.

C. Report Module

This module displays the output of the processes taken place in the algorithm module. After the data entered has been processed (encrypted or decrypted), the result of each algorithm chosen is displayed (one on each tab/page for convenience and readability). The report will include result of the encryption, the data size, the encryption parameters, computational speed, key length value, encryption ratio, resources utilized (e.g. Memory, CPU etc.) and the break rate. All these parameters will be displayed for each algorithm that was executed independently of the others (i.e. every algorithm will be executed one after the other so that each can have full system resources at its disposal). In this module, the resulting report will be logged in the database for record keeping which can later be referred to.

The result of this module is the key to determining the most secure and appropriate encryption algorithm for each data type, size and scenario.

VI. EXPERIMENTAL DETAILS

This section shows the results obtained from running the Java simulation program using different data loads. The results show the impact of changing data load on each algorithm and the impact of Cipher Mode (Encryption Mode) used.
The file name, file extension and file size of the experiment is as follows:
File name: test File extension: .mp4 File size: 26.1MB
The Full cycle test configuration was selected. Therefore, all the four algorithms (DES, 3DES, AES, and BLOWFISH) were selected. Also, all the system resources were selected. The encryption process selected was "Encrypt" (converting plaintext to cipher text).

Table 1 shows encryption memory consumption rate in megabyte of the same sizes of files of same type. From the results in Table 1 it can be find that the result for the same size of data varies proportional to the size of data file. Encryption memory consumption rate of the algorithms are not the same.
Observation: For each encryption algorithm same parameters are used for files of the same size.

Table 1: Memory consumption rate in megabyte of the algorithms.

Algorithms	Memory Consumption Rate (MB)
Blowfish	30
AES	25
3DES	16
DES	15

Table 2 shows encryption CPU utilization time period of the same sizes of files of same type. From the results in Table 2 it can be find that the result for the same size of data varies proportional to the size of data file. Encryption CPU utilization time of the algorithms are not the same.
Observation: For each encryption algorithm, same parameters are used for files of the same size.

Table 2: CPU Utilization time period in (%) x 1000.

Algorithms	CPU Usage (%)x1000
Blowfish	120.2343
AES	94.8532
3DES	85.0985
DES	96.4894

VII. RESULTS FROM EXPERIMENT

A point can be noticed here that 3DES requires always more time than DES because of its triple phase encryption characteristic. Blowfish, although it has a long key (448 bit), outperformed DES, 3DES and AES encryption algorithms. DES and 3DES are known to have worm holes in their security mechanism, Blowfish and AES, on the other hand, performed the encryption faster and with less system resources. These results have nothing to do with the other loads on the computer since each single experiment was conducted multiple times resulting in almost the same expected result.

Table 3 shows encryption speed of the same sizes of files of same type. From the results in Table 3 it can be find that the result for the same size of data varies proportional to the size of data file. Encryption speed of the algorithms are not the same.
Observation: For each encryption algorithm same parameters are used for files of the same size.

Table 3: encryption speed in milliseconds.

Algorithms	Encryption speed (ms)
Blowfish	6471.2416
AES	4947.7483
3DES	7157.5979
DES	4747.2977

Table 4 shows encryption different key size in bit of the same sizes of files of same type. From the results in Table 4 it can be find that the result for the same size of data varies proportional to the size of data file. Encryption different key size of the algorithms are not the same.
Observation: For each encryption algorithm same parameters are used for files of the same size.

Table 4: Different Key Size in bits.

Algorithms	Key length in (bit)
Blowfish	448
AES	142
3DES	147
DES	132

Table 5: Rresult from the encrypt algorithms.

File Name	File Size (KB)	Algorithm	Eval. Process	Memory (MB)	CPU Usage (%)	Real Time (ms)	Key Length (bits)
testMP4	26786	BLOWFISH	Encrypt	30	-3.902996	6471.2416	50390
testMP4	26786	AES	Encrypt	16	0.009485316	4947.7483	142
testMP4	26786	3DES	Encrypt	16	0.08509851	7151.5979	147
testMP4	26786	DES	Encrypt	15	0.09648943	4747.2977	132

Table 5 shows the result obtained from the experiment carried out using the four algorithms to test the memory consumption rate, CPU utilization time period, Encryption speed and key length of the algorithms.

Figure 3 shows that Blowfish takes highest memory for encryption, and DES take least memory for encryption, being fastest. 3DES is a trick to reuse DES implementation by cascading three instances of DES with distinct keys. 3DES is believed to be secure up to least "2^{112}" security was designed for different hardware implementation but it is less efficient in software. Blowfish consumes the highest time among all. Blowfish is efficient in software, at least on some software platforms. It uses key-dependent lookup tables; hence performance depends on how platform handles memory and caches.

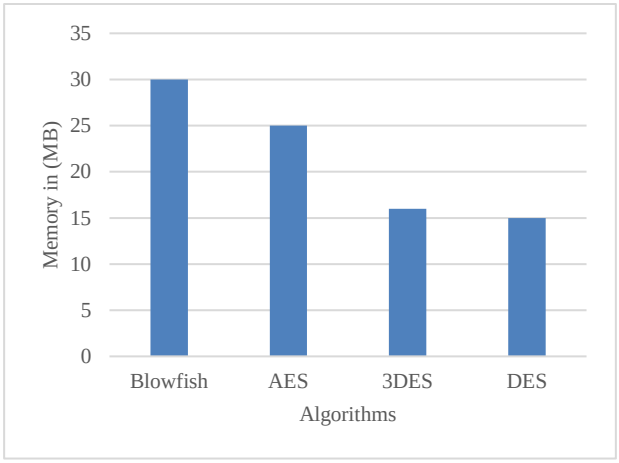

Figure 3: Graph Showing Memory Usage.

Figure 4 shows that Blowfish takes highest CPU utilization time period, and 3DES take least CPU utilization for encryption, being fastest. AES also follow 3DES in CPU utilization while DES is very close in CPU utilization to AES,

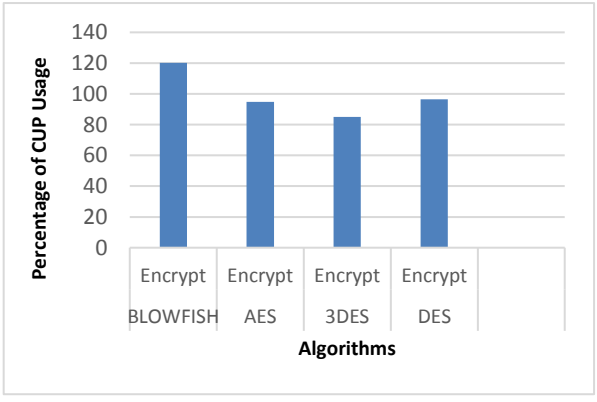

Figure 4: Graph Showing CPU Usage.

but must be noted that 3DES is a trick to reuse DES implementation by cascading three instances of DES with distinct keys.

Figure 5 shows that Blowfish exhibits highest encryption speed whereas DES exhibits least encryption speed. Encryption speed tells us the degree of diffusion of information. This shows that Blowfish will be best perform in encryption speed that others algorithms.

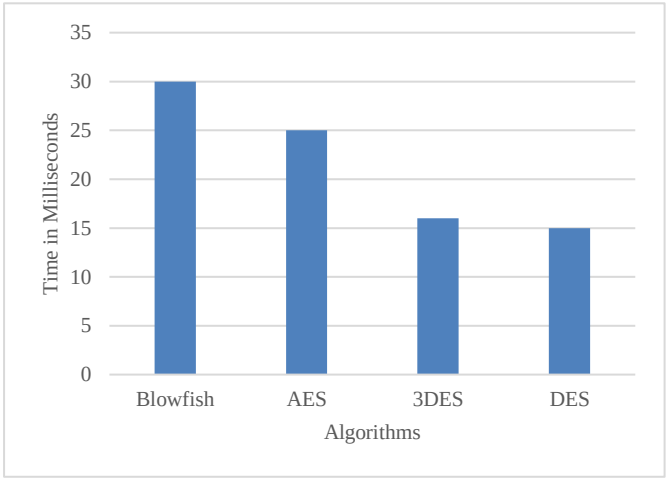

Figure 5: Graph Showing Encryption Speed of the Algorithms.

Figure 6 shows that Blowfish exhibits highest Avalanche effect whereas DES exhibits least Avalanche effect. Avalanche tells us the degree of diffusion of information. A change of one bit in plaintext leading to significant change in bits of output information AES uses a substitution permutation using multiplicative inverse and affine transformation over a gliosis field leading to high mixing of information leading to high diffusion in output.

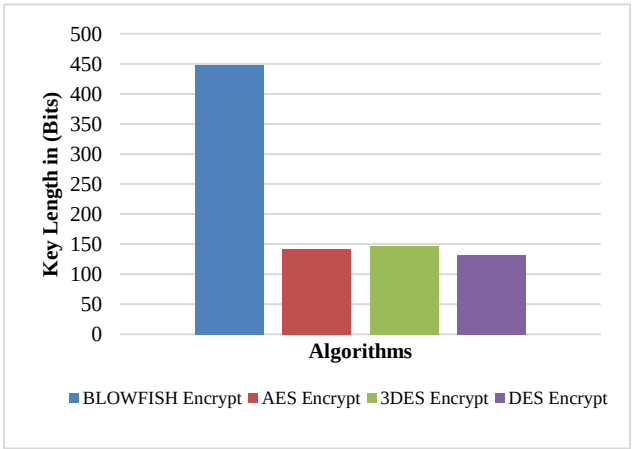

Figure 6: Graph Showing Key Length.

The Single algorithm test configuration was selected. Therefore, one algorithm (Blowfish) was selected. Also, all the system resources were selected. The encryption process selected was "Encrypt" (converting plaintext to cipher text). Blowfish, with key (448 bit) was selected alone with all the resources to be tested. Due to the increased size of the file, the algorithm spent more time and memory resources decrypting the file.

VIII. CONCLUSION

The use of cryptography to protect data by controlling accessibility to such data is an approach that have existed for long and with the great increase in how every aspect of our human endeavor's dependent on information technology. This has lead credence to need and utilization of cryptography to protect such data from the prying eyes of others.

Each of the encryption methods has its own strong and weak point. In order to apply a suitable cryptography algorithms to an application, someone should have knowledge regarding performance, strength and weakness of the algorithms. The result obtained by the application is exported to Microsoft excel for further analysis and a graph plotted in figure 3-6 for each resourced measured. The graphs and reports shows that blowfish algorithm utilizes more memory, CPU usage and time in performing its cryptographic operations which it owes to the fact that it uses a much higher key length (448bit).

It is also observed that the key length is proportional to resource utilization based on the result of the graph generated from the result of the experiments carried out. These results suggest that cryptographic algorithms with high key length should not be recommended for power and memory sensitive devices which are mostly known to be of very small size and cannot work well under heated conditions. Without the key which is what is majorly used in the decryption process, further research can be carried out to find how it can be decrypted without a pre knowledge of the key used in the encryption process.

Furthermore, power utilization can also be researched into so as to confirm the suggested result that the power utilized by the system is a function of the work done by the system. Applications of encryption in communication and how it can further protect the integrity without affecting the accessibility and availability of the information being protected.

REFERENCES

Abdul, D. S.; A. H. M. Kader and M. M. Hadhoud. (2009). Performance Evaluation of Symmetric Encryption Algorithms, Communications of the IBIMA, 8 (1): 58-64.

Agrawal, M. and Mishra, P. (2012). A Comparative Survey on Symmetric Key Encryption Techniques, International Journal on Computer Science and Engineering, 4 (2): 877-882.

Alam, M. I. and Khan, M.R. (2013). Performance and Efficiency Analysis of Different Block Cipher Algorithms of Symmetric Key Cryptography, International Journal of Advanced Research in Computer Science and Software Engineering, 3 (10):713-720.

Amit, D. (2000). Data Encryption using DES/Triple-DES Functionality in Spartan-II FPGAs, Xilinx, WP115.

Ankita, V.; G. Paramita and M. Sunita. (2016). Comparative Study of Different Cryptographic Algorithms, International Journal of Emerging Trends & Technology in Computer Science, 5 (1): 58-63.

Apoorva, K. Y. (2013). Comparative Study of Different Symmetric Key Cryptography, IJAIEM, 2 (7): 204-206.

Bala, T. and Kumar Y. (2015). Asymmetric Algorithms and Symmetric Algorithms: A Review, International Journal of Computer Applications, 1-4.

Chadi, R. and Pierre, E.A. (2015). Comparative Analysis of Block Cipher-Based Encryption Algorithms: a Survey, Information Security and Computer Fraud, 3 (1): 1-7.

Chandramouli, R. (2006). Battery power-aware encryption - ACM Transactions on Information and System Security, 9 (2): 12-25.

Dhawan P. (2002). Performance Comparison: Security Design Choices, Microsoft Developer Network, USA.

Hamdan, O. A.; B. B. Zaidan, A. A. Zaidan, A. J. Hamid, M. Shabbir and Y. Al-Nabhani. (2010). New Comparative Study between DES, 3DES and AES within Nine Factors, Journal of Computing, 2 (3): 152-157.

Hirani, S. (2003). Energy Consumption of Encryption Schemes in Wireless Devices Unpublished Thesis, university of Pittsburgh.

Idrus, S. Z. S.; S. A. Aljunid and S. M. Asi (2008). Performance Analysis of Encryption Algorithms Text Length Size on Web Browsers, International Journal of Computer Science and Network Security, 8 (1): 20-25.

Kahn, D. (1983). Secrets of the new cryptology, Macmillan. Key management, New York.

Karthik, S. and Muruganandam, A. (2014). Data encryption and decryption by using triple DES and performance analysis of crypto system, (2) (11): 2347-3878.

Krishnamurthy, G. N.; V. Ramaswamy, G. H. Leela, and M. E. Ashalatha (2008). Performance enhancement of Blowfish and CAST-128 algorithms and Security analysis of improved Blowfish algorithm using Avalanche effect", International Journal of Computer Science and Network Security, 8 (3): 1-15.

Marwaha, M.; R. Bedi, A. Singh and T. Singh (2013). Comparative Analysis of Cryptographic Algorithms, International Journal of Advanced Engineering Technology, 4 (3): 16-18.

Nadeem, A. and Javed, M. Y. (2005). A Performance Comparison of Data Encryption Algorithms, First International Conference, Information and Communication Technologies, 84- 89, USA, IEEE.

Pratap, C. M. (2012). Superiority of blowfish Algorithm. International Journal of Advanced Research in Computer Science and Software Engineering, 2 (9): 34-39.

Priya, D. (2002). Performance Comparison: Security Design Choices, Microsoft Developer Network USA.

Priyadarshini, P.; N. Prashant, D. G. Narayan and S. M. Meena (2016). A Comprehensive Evaluation of

Cryptographic Algorithms: DES, 3DES, AES, RSA and Blowfish, Procedia Computer Science 78 (1): 617 – 624.

Priyanka, A.; S. Arun and T. Himanshu (2012). Evaluation and Comparison of Security Issues on Cloud Computing Environment, World of Computer Science and Information Technology Journal, 2 (5): 179-183.

Ranjeet, M.; S. Vivek A. Jibi and M. Rajni (2015). Analysis and comparison of symmetric key cryptographic algorithms based on various file features, International Journal of Network Security & Its Applications, 6 (4): 43-52.

Pahal, R. and Skumar, V. (2013). Efficient Implementation of AES, International Journal of Advanced Research in Computer Science and Software Engineering, 3 (7): 19-23.

Saini, B. (2014). Survey on Performance Analysis of Various Cryptographic Algorithms, International Journal of Advanced Research in Computer Science and Software Engineering, 4 (4): 1-4.

Sandeep, S. (2012). Introduction to cryptography. Available online at: http:www.scribd.com/doc/83051190/ introduction-to-crytograpgy.html Access on November 16, 2015.

Seth, S. M. and Mishra, R. (2011). Comparative analysis of Encryption algorithm for data communication, International Journal of Computer Science and Technology, 2 (2): 292-294.

Singh, G.; A. Kumar and K. S. Sandha (2005). A Study of New Trends in Blowfish Algorithm, International Journal of Engineering Research and Applications, 1 (2): 321-326.