A Dynamic Round Triple Data Encryption Standard Cryptographic Technique for Data Security

Oluwatobi Noah Akande[1(✉)], Oluwakemi Christiana Abikoye[2], Aderonke Anthonia Kayode[1], Oladele Taye Aro[3], and Oluwaseun Roseline Ogundokun[1]

[1] Department of Computer Science, Landmark Univeristy, Omu-Aran, Kwara, Nigeria
akande.noah@lmu.edu.ng
[2] Department of Computer Science, University of Ilorin, Ilorin, Kwara, Nigeria
[3] Department of Mathematical and Computing Sciences, Kola Daisi University, Ibadan, Nigeria

Abstract. Cryptographic techniques have been widely employed to protect sensitive data from unauthorized access and manipulation. Among these cryptographic techniques, Data Encryption Standard (DES) has been widely employed, however, it suffers from key and differential attacks. To overcome these attacks, several DES modifications have been proposed in literatures. Most modifications have focused on enhancing DES encryption key; however, the strength of a cryptographic technique is determined by the encryption key used and the number of encryption rounds. It is a known fact that Advanced Encryption Standard (AES) cryptographic technique with 14 encryption rounds is stronger than AES with 12 rounds while AES with 12 rounds is stronger than AES with 10 rounds. Therefore, this study proposed a DES cryptographic technique whose number of rounds is dynamic. Users are expected to specify the number of encryption and decryption rounds to be employed at run time. Moreover, a predefined number of shifting operations which is left circular shift 2 was chosen for each encryption round. As, a trade-off in complexity, the number of Substitution box (S-box) was also reduced to 4, so that the input to the S-boxes would be arranged in four 12-bit blocks for the X-OR operation and not six 8-bit blocks as in the traditional DES. Finally, three keys were used to encrypt, decrypt and encrypt the plaintext ciphertext as in triple DES. The modified DES yielded a better avalanche effect for rounds greater than 16 though its encryption and decryption time were greater than that of the traditional DES.

Keywords: Data Encryption Standard (DES) · Modified DES · Electronic medical information · Data and information security

O. Gervasi et al. (Eds.): ICCSA 2020, LNCS 12254, pp. 487–499, 2020.
https://doi.org/10.1007/978-3-030-58817-5_36

1 Introduction

The huge volume of data available on the internet has made data and information security a topmost issue of concern for 21st century researchers. With an internet user increase of 50.3%, 15.9% and 11.5% in Asia, Europe and Africa respectively for the first quarter of 2020, the amount of data available on the internet is enormous [1]. Besides data available on the internet, it is estimated that 88% of businesses in US have over 1 million folders that are unprotected and freely accessible to employers [2]. Statista submitted that data breaches in the US have been increasing annually from 783 instances in 2014 to 781 cases in 2015; 1093 instances in 2016 to 1579 cases in 2017; 1244 instances in 2018 to more than 3800 cases in 2019 [3]. These have put US at the index position when it comes to data breaches. Furthermore, the frequency and complexity of data and cybersecurity breaches are on the increase, with 8% of government parastatals, 29.2% of medical and healthcare institutions and 45.9% of businesses being affected in 2018 alone [4]. Every data and cybersecurity breaches come with a cost. The global average cost of data breach in 2019 was put at $3.92 million in contrast to $3.86 million recorded in 2018 and $3.62 recorded in 2017 [5]. With these statistics, regardless of how simple it may be, organizations need to implement at least one security technique to ensure the confidentiality, maintain the integrity and guarantee the availability of their sensitive data and information from unauthorized access, manipulation or theft.

Cryptographic techniques have been widely used to achieve these. Cryptography makes data unreadable and unmeaningful to a third party, thereby making them unsusceptible to threats or attacks. Majorly, cryptography intends to maintain privacy of data so that only intended recipients will be able to read the messages; it can also be used to verify the identity of senders or recipients (Authentication), prove that a particular message has not been read or tampered with therefore assuring that the received message is the same with the original message (integrity). Furthermore, cryptographic techniques can be used to affirm the source of the received message (Non-repudiation). Generally, cryptographic techniques require a secret key for encrypting or decrypting sensitive messages. This could be used to categorize them to symmetric or asymmetric cryptographic techniques. Symmetric cryptographic techniques are also called secret key cryptography while asymmetric cryptographic techniques are called public key cryptography. While symmetric uses the same key for encryption and decryption, asymmetric uses different keys for encryption and decryption. Therefore, symmetric cryptography guarantees the privacy and maintain the confidentiality of messages. In the same vein, asymmetric cryptography enforces authentication and maintain non-repudiation of messages. In addition to symmetric and asymmetric cryptographic techniques, hash function is another category where data are mathematically transformed into an irreversible cipher text; this connotes that the plain text cannot be retrieved from the cipher text. They are majorly used to maintain the integrity of messages sent.

Symmetric cryptography could further be categorized as a stream or block ciphers. Stream ciphers work on a single bit per time but uses a feedback mechanism to constantly change the key. However, a block cipher converts a plaintext into a block of

data and carries out the encryption one block per time without changing the key used on each block. The same plaintext in a block cipher will produce the same ciphertext whereas different ciphertext will be generated from the same plaintext in a stream cipher [11]. Block ciphers could be implemented in several modes such as: cipher feedback mode, output feedback mode, counter mode, cipher block chaining mode, counter mode and electronic codebook mode being the simplest. As illustrated in Fig. 1, several symmetric and asymmetric cryptographic techniques have been proposed and employed for various security applications over the years [6–8]. However, this article focuses on Data Encryption Standard (DES) cryptographic technique.

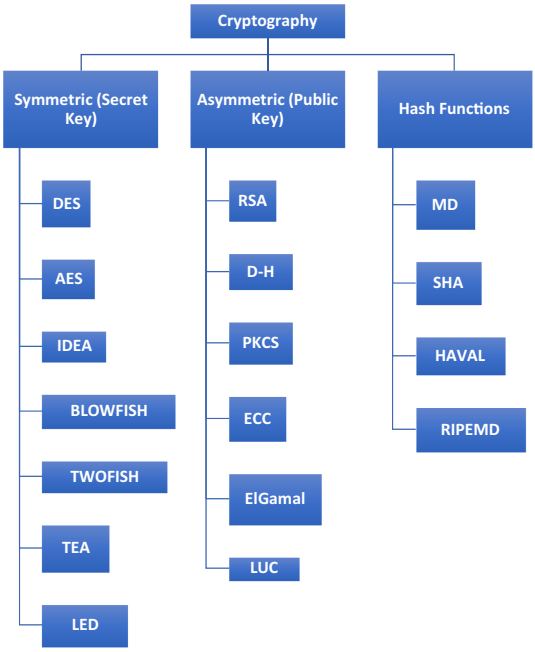

Fig. 1. Overview of cryptographic techniques

Where IDEA is International Data Encryption Algorithm, TEA is Tiny Encryption Algorithm, LED is Light Encryption Device, RSA was named after its inventors' authors Ronald Rivest, Adi Shamir, and Leonard Adleman, D-H is Diffie-Hellman, DSA is Digital Signature Algorithm (DSA), ECC is Elliptic Curve Cryptography, PKCS is Public Key Cryptography Standards, MD is Message Digest, SHA is Secure Hash Algorithm, RIPEMD is RACE Integrity Primitives Evaluation Message Digest, HAVAL is HAsh of VAriable Length. The next section of this article explains the encryption process of DES and Triple DES algorithm. An overview of several modifications to DES that have been reported in literature is provided under the related works section while the detailed steps taken to achieve the proposed DES modification is provided in the methodology section. The proposed modification was evaluated on

an electronic medical database as explained in the results and discussion section. Processing time and Avalanche effects were used as performance evaluation metrics. Conclusion and recommendation for future studies were provided in the concluding part of the article.

2 DES/Triple DES Algorithm

DES is a 64-bit block cipher symmetric cryptographic algorithm that uses 56-bit key for encryption and decryption. As a 64 bits block cipher, DES divides input data into blocks of data and encrypts 64 bits of data at a time. DES is flexible in nature therefore it can operate in any of the cryptographic modes. Like other cryptographic algorithms, DES uses permutation and substitution operations for its encryption and decryption process. Substitution entails replacing a value with another while permutation reorders the positions of the bits in the input data. These processes are repeated in a number of times called rounds and it is generally believed that the more the number of rounds the higher the strength of the algorithm. DES encryption process goes thus:

Step 1: convert the input plaintext and the key into decimal and hexadecimal form using the ASCII character table
Step 2: afterwards, convert the decimal and hexadecimal form into binary bits
Step 3: divide the binary bits into 64 bits block size
Step 4: use the Initial Permutation (IP) table to randomize the bits in plaintext data block to yield E0
Step 5: substitute the key bits using permuted Choice 1 (PC - 1) table.
Step 6: divide the resulting bits into two; represent the leftmost 28 bits by L0 and the rightmost 28 bits by R0.
Step 7: using the cipher F function, shift Li and Ri to the left with "i" being the encryption round.
Step 8: combine Li and Ri from every round to produce a 56 bits LiRi
Step 9: using permuted Choice 2 (PC – 2) table, randomize LiRi to produce variable Ki
Step 10: starting with E0, use the expansion table to expand 32bits of the data to produce a new 48bits data Ei where i = i – 1.
Step 11: Perform modular X-OR of Ei and Ki to produce a 48bits Ai
Step 12: divide Ai into eight blocks data of six bits each.
Step 13: replace each of the eight blocks in step 12 with corresponding value in the substitution block to give a new variable Ni
Step 14: using the P – Box table, permutate each bit of Ni for i = 0 to 15
Step 15: XOR Ni and Ei-1 to produce a 32bits variable Ci
Step 16: after completing the 16 rounds, merge C_{16} and E_{16} to produce 64 bits $C_{16}E_{16}$
Step 17: Reverse positions of $C_{16}E_{16}$ to produce $E_{16}C_{16}$
Step 18: using Inverse Initial Permutation table, permutate $E_{16}C_{16}$ to produce the expected 64 bits Ciphertext

Steps 5 to 16 will be iterated 16 times where i indicates the current round number. To decrypt the ciphertext, steps 18 to step 1 will be performed backwards. A graphical representation of the encryption process is provided in Fig. 2.

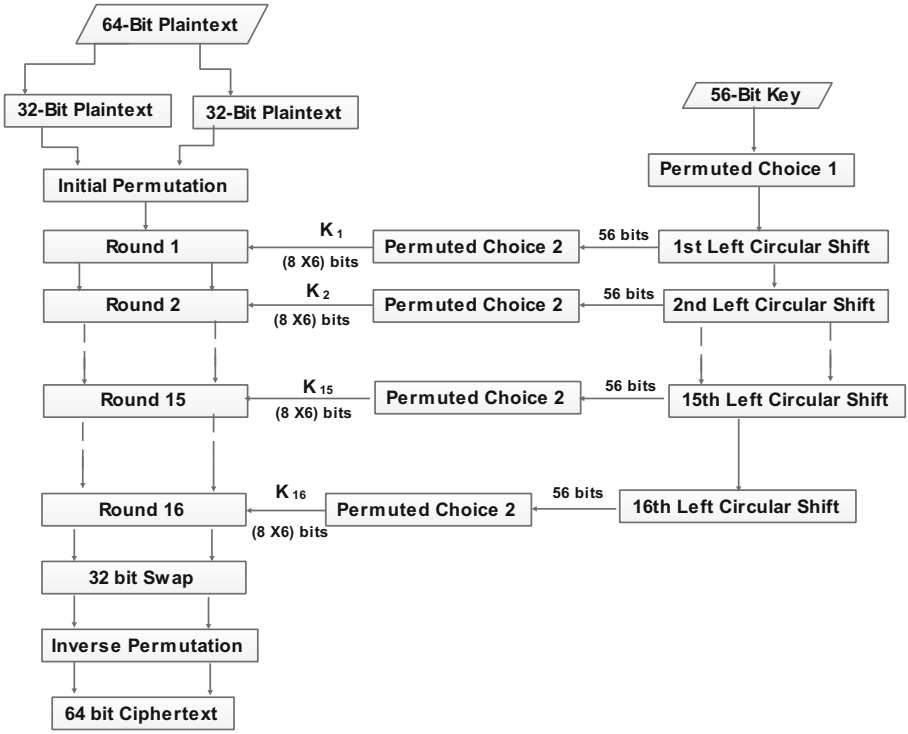

Fig. 2. DES encryption process

DES is vulnerable to key attack therefore its strength is dependent on the type of key used. To make DES more resistive to key attack, triple DES was introduced. It uses three different keys to encrypt, decrypt and encrypt each block of data. This means that the total length of keys used for the first encryption, decryption and the final encryption is 168 bits. These sub-keys can be used in three ways as opined in [14]. The three sub-keys can have the same combination or different combinations and the first and second key can have different combination while the first and third key can have the same combination. However, when the three sub-keys have different combinations then the effective key length is 168 bits but when the three sub-keys have the same combination, the effective key length is 56 bits (which is considered as the weakest combination). The second option where the first and third key have the same combination has an effective key length of 112 bits which is still preferable.

3 Related Works

The major strength of DES is its weak 58 bit encryption key which has made it susceptible to diverse forms of attack. The high computational power of 21st century computers has also favour most of these attacks. Therefore, several works aimed at improving the key generation process of DES has been proposed in the literature. For instance, a two-stage dynamic key generation approach was proposed in [10]. Linear Feedback Shift Register (LFSR) was used to generate the first key while chaotic encryption was used to generate the second key. LFSR uses a linear feedback function and a shift register for its key generation process while chaotic encryption uses a One-dimensional Logistic map for its key generation. Furthermore, two variants of DES: Dynamic DES (DDES) and Hashed DES (HDES) were proposed in [13]. DDES uses seed generator, Pseudo-Random Generator (PRG), boxes generator, seed filter and seed distributor to organize the relationship between the S-boxes and the generated P-box arrangements during each round of the encryption and decryption process. The seed generator uses three encryption keys to generate an initial seed that is used by the PRG to produce random numbers. These numbers are used as input into the boxes generator that produces the S-boxes and P-box arrangements dynamically for each round. After the 16 encryption rounds, the final seed generated is inserted into the ciphertext by the seed distributer. For the decryption process, the seed filter is used to extract the seed embedded in the ciphertext. On the contrary, a hash function is used by HDES to generate a random fingerprint for each blocks of data. The seeds produced by the fingerprint are then used to securely select S-boxes for each encryption and decryption rounds. The degree of randomness of the proposed techniques as measured by chi-square revealed that DDES with 93.7% has the best degree of randomness followed by 3DES with 95.9%, HDES with 97.7% and DES with 98.3%. Furthermore, hamming distance was used to measure the degree of randomness of the proposed encryption techniques. Results obtained revealed that DDES has the highest hamming, followed by HDES, then DES and 3DES. Nevertheless, 3DES has the highest encryption processing time while DDES has the highest encryption time.

The F-function of DES encryption is used to carry out XOR operation between the encryption key and the input plaintext. However, this function was replaced with striding and filtering techniques in [9]. After, the expansion process, a 48bit filter in form of a matrix is divided into four 3 by 4 matrices. The output of the expansion process was then XORed with the first filter. The result was then XORed with the second, third and fourth filter sequentially. The final result was then fed into the substitution box for the next encryption stage. An avalanche effect of 55% was recorded by the proposed modification. To introduce an additional degree of confusion to DES, authors in [16] integrated odd-even substitution process into the encryption process. After converting the plaintext into binary value, the even positions are replaced by 1 while the odd positions are replaced by 0. The modified DES recorded an encryption time of 365.2 ms while the traditional DES achieved an encryption time of 355.8 ms. Similarly, authors in [14] proposed a DES and 3DES cryptographic technique for securing data stored in smartcards. Simulation results revealed that data writing using DES encryption is faster than that of 3DES while data reading using DES

decryption is also faster than 3DES decryption. Also, the execution time of the encryption process is faster than that of the decryption process. Towards increasing the performance of 3DES cryptographic algorithm implemented in ECB mode, a 48-stage pipelined depth design was proposed in [15]. Traditional DES is known to employ two permutations and 16 rounds of Feistel functions for its encryption and decryption process. So, to pipeline the DES, extra registers were added to the Feistel function rounds, the key bank and the key scheduler. Also, right rotations were integrated into the decryption key scheduler and finally a key bank was used to buffer the keys used for the 15 and 31 cycles. A high throughput of 3.2 Gbps at 50 MHz clock was recorded, however, a high cost was incurred due to the extra registers added to each DES component. Similarly, time variable sub-keys was combined with a 16 stage pipelining design in [12]. Different sub-keys that are dynamically generated by the permutation choice 1 box are used to encrypt the plaintext at each round. It is believed that the time variant behavior of the key will make the proposed technique difficult to break by hackers. The proposed design achieved a higher throughput value when compared to similar pipelining designs.

Towards further strengthening DES, its key length and substitution box was enhanced in [17]. DES key length and block size were increased from 64 to 128 bits while S-box values were increased from 64 to 256. In addition, if a computer with high computational power is programmed to decrypt a cipher at 50 billion keys per second, then it will take 400 days to break traditional DES, 800 days for a 112 bits 3DES and 5 $\times$ 1021 years for AES as well as the proposed DES. The proposed DES also yielded a higher avalanche effect when compared to the traditional DES. In wireless telecommunications system, it has been observed that the higher the signal to noise ratio of a wireless channel the lower the higher the accuracy of the received ciphertext. However, low SNR will always lead to loss of the ciphertext bits. Therefore, authors in [18] proposed a modification to DES that is aimed at minimizing its bit error rate in wireless applications. To achieve this, a new round with a new 80-bit key was added to the S-boxes. Also, the modified algorithm accepts 64 bits plaintext like the traditional DES but 128 bits ciphertext is produced instead of 64 bits in traditional DES. The modified DES also uses 136 bits key for encryption and decryption instead of 64 bits used by the traditional DES. This made it more secured to brute force and differential cryptanalysis attacks. To evaluate the performance of the proposed technique, several bits of errors were introduced into the input blocks of data. When 7 bits of error was introduced, a bit error rate of 0.22 was recorded as against 0.5 recorded with the traditional DES. However, when the complexity of the modified DES was measured in terms of clock cycles needed to encrypt each block, it was observed that traditional DES requires 160 byte-wise OR operations, 176 byte-wise shift operation, and 320 byte-wise AND operations to encrypt each block. This connotes that 8136 clock cycles were needed by traditional DES to encrypt each block while the modified DES requires 8944 clock cycles. Furthermore, though the encryption time of both algorithms increases with an increase in file size, the encryption time of the modified DES is higher.

4 Methodology

The strength of cryptographic algorithms is dependent on the encryption key used and the number of rounds required during its encryption process. It is generally believed that the more the encryption rounds the stronger the cryptographic algorithm. For instance, 128, 192- and 256-bits AES algorithm require 10, 12 and 14 encryption rounds respectively while DES algorithm requires 16 encryption rounds. Therefore, to further strengthening DES algorithm, a dynamic round DES algorithm is presented. Here, the number of rounds for the encryption and decryption will be specified at runtime by the user. However, a predefined number of shifting operations which is left circular shift 2 was chosen; this would be utilized for each round. Also, as a trade-off in complexity, the number of s-boxes used was reduced to 4, so that the input to the 4s-boxes would be arranged in four 12-bit blocks for the X-OR operation. Finally, three keys were used to encrypt, decrypt and encrypt the plaintext ciphertext as in triple DES. A graphical illustration of the modified DES is provided in Fig. 3 while the corresponding pseudocode goes thus:

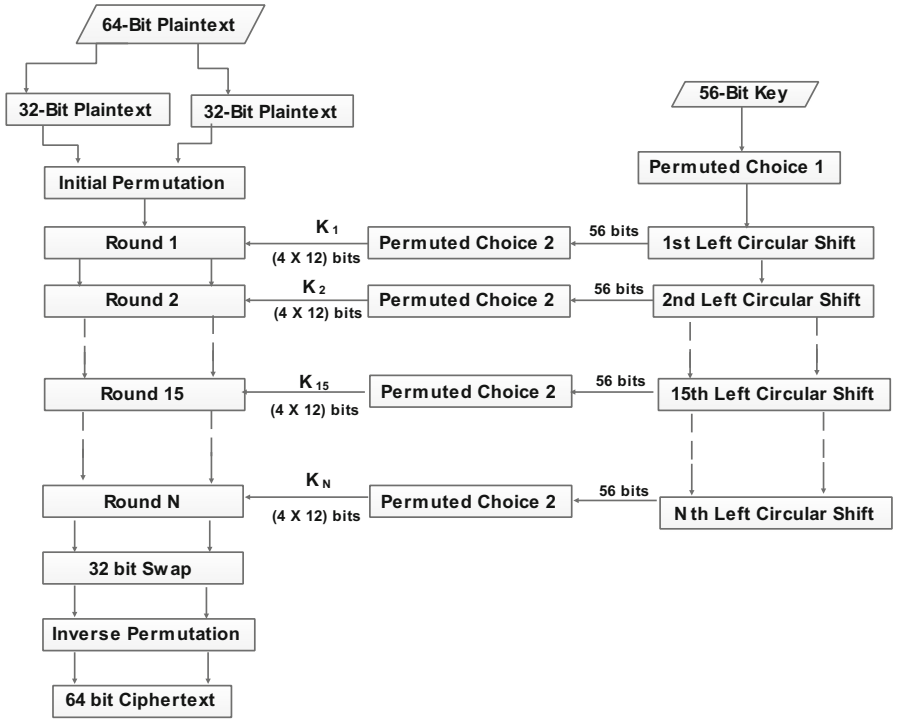

Fig. 3. Modified DES encryption process

Step 1: Input data is divided into 64 bits block size
Step 2: Permute with IP-1 table
Step 3: Divide into two 32 bits left and right half
Step 4: Initialize 'N' round operations applying cipher F function (with reduced number of s-boxes) with auto-generated key on right half.
Step 5: Perform modular X-OR of left half 32 bits with result from step 4 which be comes new right half 32bits.
Step 6: Initial right half 32 bits becomes new left half bits
Step 7: Combine the 32bits from step 5 and step 6
Step 8: Perform inverse permutation to obtain cipher text.
Step 9: For decryption, perform backwards from step 8 to step 1 to obtain plain text
Step 10: Repeat step 1–8 twice with same key as input and step 8 to step 1 backwards with separate key to perform the triple DES encryption. i.e. EncK1 (DecrK2) EncK1

5 Results and Discussion

5.1 The Developed Electronic Medical Information (EMI) System

An Electronic Medical Information (EMI) system was developed to evaluate the encryption and decryption capability of the proposed cryptographic technique. The medical database keeps sensitive textual and graphical information of patients in a hospital. After entering the needed patient information as shown in Fig. 4, the medical personnel are expected to choose an encryption round which is expected to be known to other legitimate users of the EMI system.

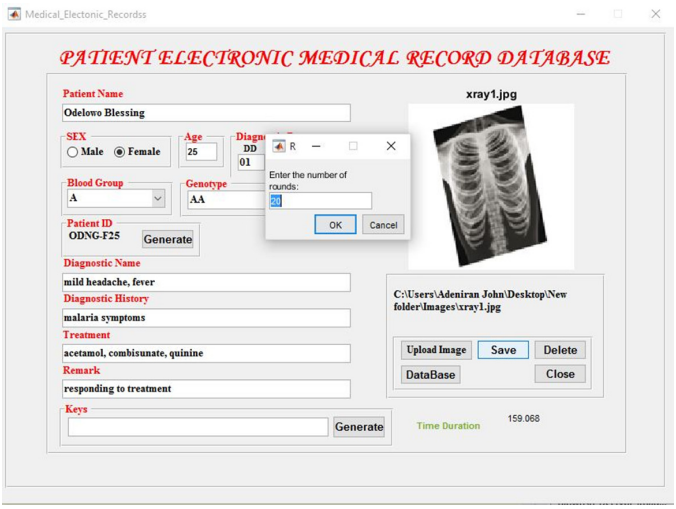

Fig. 4. EMI requesting for the number of encryption rounds

For the decryption process, the user is expected to automatically generate the encryption key and provide the correct number of encryption rounds. Should a wrong number of encryption round be supplied, a scrambled information will be displayed as shown in Fig. 5. A correct number of round will decrypt the encrypted information as displayed in Fig. 6.

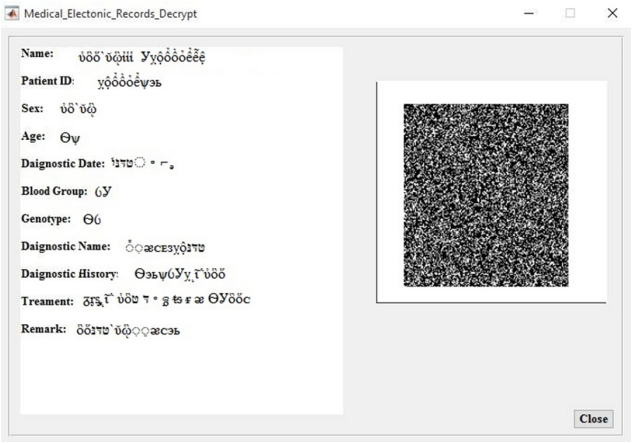

Fig. 5. Output due to wrong number of rounds or decryption key

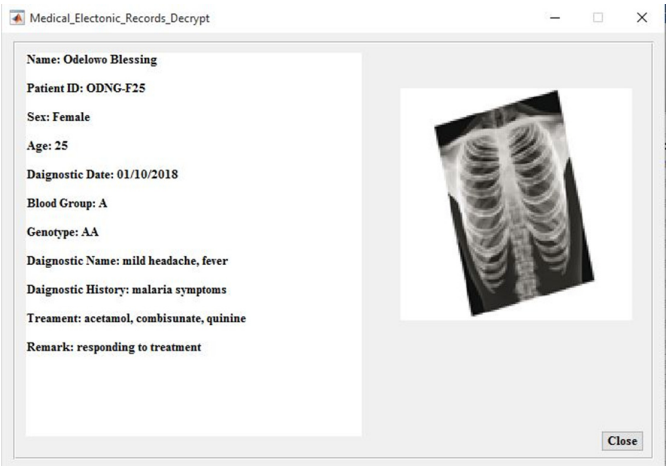

Fig. 6. Decrypted patient information

Table 1. Processing time of the modified triple DES

Number of rounds	Encryption time (sec)	Decryption time (sec)
10	57.4009	49.0655
11	62.8198	58.4639
12	66.8347	62.0624
13	73.0785	70.5850
14	72.3833	70.4269
15	77.5545	76.1033
16	80.8605	77.1530
17	89.7312	87.7793
18	96.2235	87.8725
19	95.8044	91.9472
20	102.217	96.3151

Table 2. Avalanche effects of modified triple DES

No of rounds	Plain text	Cipher text	Avalanche effects (%)
15	97157A6FC8**E**4BBE4	AB17796FC2E4240C	8(50)
	97157A6FC8**F**4BBE4	A710386FCAF407A0	
16	50ED00C48388EA9**B**	BCD0C1F84644BE33	14(87.5)
	50ED00C48388EA9**A**	62E5C2560F22B464	
17	0FB**7**C204C2C12D39	3BBC0214C0C1A1E5	15(93.75)
	0FB**6**C204C2C12D39	CB6A207C271225D8	

5.2 Performance Evaluation

To measure the performance of the proposed technique, the processing time of the system was measured in terms of encryption and decryption time. This was done on a 64 bit, dual core HP laptop with 4 GB RAM. The number of rounds was varied in steps of one from 10 to 20 as presented in Table 1. It was observed that the encryption and decryption time increased with an increase in number of rounds. Also, with 16 rounds, the processing time of the traditional DES is lower than that of the modified DES. But in real life scenario, processing time could be traded off for security. People are interested in securing their sensitive documents regardless of the time it takes to secure the documents.

Furthermore, the avalanche effect of the proposed technique was measured as documented in Table 2. This measured the degree of diffusion of a cryptographic technique. It is used to guarantee that any little change in the input plain text will have a corresponding huge change in the cipher text. The avalanche effect increased with an increase in the number of rounds. This connotes that the degree of confusion increased with an increase in the number of rounds. This is actually a good result. However, the avalanche effect of the modified DES with 16 rounds was evaluated with the traditional DES with 16 rounds. A higher avalanche effect of 90.43% was recorded by traditional DES against 87.50% achieved by the modified DES. However, higher rounds yielded a higher avalanche effect as presented in Table 2.

6 Conclusion

Our increasing sensitive data and information need to be protected from unauthorized access, manipulation and alteration. The effect of a small breach of data could cost an organization a huge amount of money and undermine the confidence reposed in their services. It is advisable for organization to implement at least one cryptographic technique to protect their sensitive data. This article has presented a modified DES cryptographic algorithm that could be used to secure sensitive data and information. Several literatures have reported modifications to the encryption key of traditional DES; however, this article explored the possibility of making the encryption rounds of DES user dependent. With this arrangement, users are expected to choose the encryption rounds desired. This encryption is expected to be greater than the default 16 rounds. Though the modification leads to an increase in encryption and decryption time, the performance evaluation results carried out revealed that the technique's avalanche effect increases with an increase in encryption rounds. The higher the avalanche effect, the greater the assurance that any small modification to the plaintext will have a huge noticeable difference in the ciphertext.

References

1. Internet Users Distribution in the World -2020 Q1. http://www.internetworldstats.com. Accessed 24 Mar 2020
2. Data under Attack: 2018 Global Data Risk Report from the Varonis Data Lab. https://info.varonis.com/hubfs. Accessed 24 Mar 2020
3. Cybercrime: number of breaches and records exposed 2005–2019. https://www.statista.com/statistics/273550/data-breaches-recorded-in-the-united-states-by-number-of-breaches-and-records-exposed/. Accessed 24 Mar 2020
4. Biggest Data Breach Statistics. https://www.digitalinformationworld.com/2019/08/biggest-data-breach-statistics.html. Accessed 24 Mar 2020
5. Cost of a Data Breach Report https://securityintelligence.com/posts/whats-new-in-the-2019-cost-of-a-data-breach-report/. Accessed 24 Mar 2020
6. Abikoye, O.C., Adeshola, G.Q., Akande, N.O.: Implementation of textual information encryption using 128, 192 and 256 bits advanced encryption standard algorithm. Ann. Comput. Sci. Ser. **15**(2), 153–159 (2017)
7. Abikoye, O.C., Haruna, A.D., Abubakar, A., Akande, O., Asani, E.O.: Modified advanced encryption standard algorithm for information security. Symmetry **11**(1484), 1–17 (2019). https://doi.org/10.3390/sym11121484
8. Akande, N.O., Abikoye, C.O., Adebiyi, M.O., Kayode, A.A., Adegun, A.A., Ogundokun, R. O.: Electronic medical information encryption using modified blowfish algorithm. In: Misra, S., et al. (eds.) ICCSA 2019. LNCS, vol. 11623, pp. 166–179. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-24308-1_14
9. Amorado, R.V, Sison, A.M., Medina, R.: Enhanced data encryption standard (DES) algorithm based on filtering and striding techniques. In: 2nd International Conference on Information Science and Systems, pp. 252–256 (2019)
10. Gautam, A.: FPGA Implementation of dynamic key generation to enhance des algorithm securities **4**(01), 673–677 (2015)

11. Kessler, G.C.: An overview of cryptography. http://Www.Garykessler.Net/Library/Crypto.Html. Accessed 20 Mar 2020
12. Oukili, S., Bri, S.: High throughput FPGA implementation of data encryption standard with time variable sub-keys **6**(1), 298–306 (2016)
13. Qasaimeh, M., Al-qassas, R.S.: Randomness analysis of DES ciphers produced with various dynamic arrangements. J. Comput. Sci. Original **13**(12), 735–747 (2017). https://doi.org/10.3844/jcssp.2017.735.747
14. Ratnadewi, B., Roy, P.A., Yonatan, H., Saleh, A., Setiawan, M.I.: Implementation cryptography data encryption standard (DES) and triple data encryption standard (3DES) method in communication system based near field communication (NFC). J. Phys: Conf. Ser. **954**, 1–9 (2018)
15. Rosal, E.D., Kumar, S.: A fast FPGA implementation for triple DES encryption scheme. Circ. Syst. **8**, 237–246 (2017). https://doi.org/10.4236/cs.2017.89016
16. Sison, A.M., Tanguilig, B.T., Gerardo, B.D., Byun, Y.-C.: Implementation of Improved DES Algorithm in Securing Smart Card Data. In: Kim, T., Ramos, C., Kim, H.-K., Kiumi, A., Mohammed, S., Ślęzak, D. (eds.) ASEA 2012. CCIS, vol. 340, pp. 252–263. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-35267-6_33
17. Sivakumar, T.K., Sheela, T., Kumar, R., Ganesan, K.: Enhanced secure data encryption standard (ES-DES) algorithm using extended substitution box (S-Box). Int. J. Appl. Eng. Res. **12**(21), 11365–11373 (2017)
18. Zibideh, W.Y., Matalgah, M.M.: Modified data encryption standard encryption algorithm with improved error performance and enhanced security in wireless fading channels, pp. 565–573 (2015)