Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X
IMPACT FACTOR: 7.056

IJCSMC, Vol. 12, Issue. 4, April 2023, pg.24 – 41

Triple Rounds to Protect Text Files

Akram Naser Adam Mousa[1]; Talal Bukewitin[2]; Amna M.M. Salem[3]; Prof. Ziad Alqadi[4]

[1]Higher Institute for Science and Technology Bayda, Libya
[2]Libya College of Computer Technology Zawia, Libya
[3]Higher Institute for Science and Technology Bayda, Libya
[4]Albalqa Applied University, Jordan

DOI: https://doi.org/10.47760/ijcsmc.2023.v12i04.003

Abstract:

Text file may be confidential and it can contain secret information that required protection from being hacked. In this research paper a simple method of text files cryptography will be introduced, this method will suit any text file with any size; the sequence of required operations will remain the same without any change even if we change the text file and the private key. The introduced method of data cryptography will provide a high level of security; the private key will provide a huge key space making the hacking process impossible. The private key will have a complicated structure with multiple components, these components are to be used to generate three keys needed to apply three rounds of encryption-decryption, the keys will be very sensitive to any minor changes, any changes in the private key components in the decryption phase will be considered as a hacking attempt. The private key will base on chaotic logistic map model to generate 2D chaotic keys. The proposed method will be implemented and the obtained results will be analyzed using various types of analysis to prove the efficiency enhancement provide by the proposed method.

Keywords: Cryptography**,** CLMM, PK, key, MSE, PSNR, CC, throughput.

Introduction

Encryption in cyber security means converting data from a readable format to an encrypted format [45-50]. The encrypted data can only be read or processed after it has been decrypted [16-20]. Encryption is the basic building block of data security. It is the simplest and most important way to ensure that computer system information is not stolen or read by someone who wants to use it for malicious purposes. Data security encryption is widely used by individual users and large companies for the purpose of protecting user information transmitted between the browser and the server. This information may include anything from payment data to personal information [35-44]. A data encryption program, also known as an "encryption algorithm" or simply "encryption", is used to develop a cipher

scheme that can theoretically only be penetrated by massive computing power [48-53]. When information or data is shared over the Internet, it passes through a series of network devices around the world, which are part of the Internet as a whole. Through the transmission of data over the public Internet, there is a possibility that it will be hacked or stolen by hackers. To prevent this, users may install certain software or hardware to ensure the secure transmission of data or information. These operations in the field of network security are known as "encryption" [40-50]. Encryption involves converting plain text that is readable to humans into unintelligible text, which is known as cipher text. This generally means taking readable data and changing it so that it appears randomly. Encryption involves the use of an encryption key, which is a set of mathematical values agreed upon by both the sender and the receiver. The recipient uses the key to decrypt the data and return it to readable plain text [25-35]] -. The more complex the encryption key, the more secure the encryption, as third parties are less likely to be able to decrypt it through brute force attacks (i.e. trying random numbers until the correct combination is guessed). Encryption is also used to protect passwords. Password encryption methods hash your password to become unreadable by hackers [35-40].

Data cryptography has the following benefits [45-53]:

- Encryption helps maintain data integrity: Hackers don't just steal information; they can even alter the data to commit fraud. While it is possible for skilled hackers to alter the encrypted data, recipients of the data will be able to detect the damage, allowing for a quick response.
- Cryptography helps organizations comply with regulations: Many industries, such as financial services or health care, have strict regulations about how consumer data is used and stored. Encryption helps organizations meet and ensure compliance with these standards.
- Encryption protects data as it travels across devices: Most of us use multiple devices in our daily lives, and transferring data from one device to another can have some risks. Encryption technology helps protect data across devices, even during transmission. Additional security measures, such as advanced authentication, also help deter unauthorized users from accessing.
- Encryption helps when transferring data to cloud storage: More and more users and organizations are storing their data in the cloud, which means cloud security is a must. Encrypted storage helps keep that data private. Users must ensure that data is encrypted during transmission, during use, and during storage.
- Encryption helps organizations secure offices: Many organizations include remote offices, especially in the post-pandemic era. This can pose cyber security risks as data is accessed from many different locations. Here, encryption helps protect against theft or accidental loss of data.
- Data encryption protects intellectual property: DRM systems silently encrypt data, in this case intellectual property such as songs or software, to prevent reverse engineering and unauthorized use or reproduction of copyrighted material.

Text files cryptography [21-23]usually done using one private key (PK), to increase the level of security and to provide a high degree of text file protection we will use three complicated PKs as shown in figure 1.

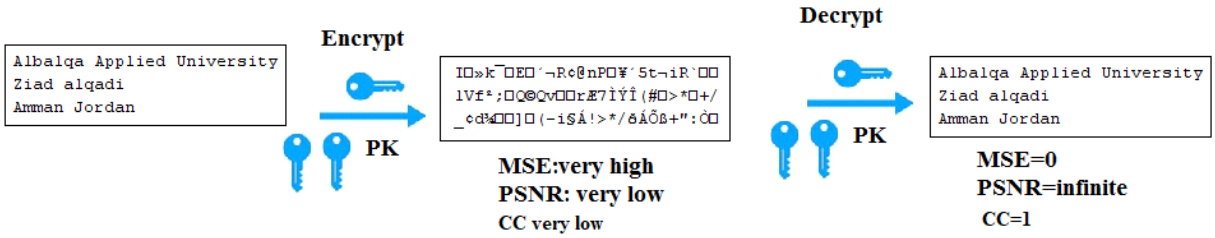

Figure 1: Text file cryptography

Text file cryptography is considered s a good method of data cryptography if it satisfies the following features[24-32]:

- **Cryptography quality**: Here the encryption process must destroy the input text file making it corrupted and un readable. The degree of destruction can be measured by mean square error (MSE) between the input file and the encrypted one, peak signal to noise ratio (PSNR) between the input file and the encrypted one, and correlation coefficient (CC) between the input file and the encrypted one. Here MSE must be very high, PSNR must be very low and CC must be very low (closed to zero).
  In the decryption phase the input file must be recovered and the decrypted file must be identical to the input file, here MSE must be zero, PSNR must be infinite and CC must equal 1. MSE, PSNR and CC are to be calculated using equations 1, 2 and 3.

$$MSE_x = \frac{1}{N}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}[S(i,j) - R(i,j)]^2 \ , N = m * n \quad \textbf{(1)}$$

$$PSNR = 10 * \log_{10}\frac{(MAX_I)^2}{MSE_t} \quad \textbf{(2)}$$

S and R are text files

$$\text{CC} = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum (x_i - \bar{x})^2 \sum (y_i - \bar{y})^2}} \quad \textbf{(3)}$$

Where:

r = correlation coefficient

x_i = values of first message

$\bar{x}$ = mean of x

y_i = values of second message

$\bar{y}$ = mean of y

- **The method must be highly secure**: The method security can be achieved by the complicity of the PK, here the key space must be very huge and the PK key must be very sensitive to any minor changes, any minor change in the PK during the decryption phase must be considered as a hacking attempt and the produced decrypted file must be corrupted.

- **The method must be efficient**: This can be achieved by minimizing both the encryption and decryption times, this will lead to maximizing the method throughput (bytes encrypted/decrypted per second).

- **Simplicity:** The method must be easy to implement and changing the methods inputs does not lead to any change in the process of encryption/decryption.

Related Work

A lot of text files cryptography methods were introduced, many of these methods were based on some standards [8-12] like data encryption standard (DES), 3DES, advanced encryption standard (AES) and blowfish (BF). These methods provided good quality parameters (MSE, PSNR and CC) and they are good to encrypt-decrypt small data sizes, increasing the text file size will increase rapidly both the encryption and decryption times, thus the methods throughput will drop down. The proposed standard based methods shared a lot of features, some of these features can be considered as disadvantages: These methods used a fixed length PK and this key was used to generate other needed sub keys, the data to be encrypted-decrypted was divided into equal in size blocks, block size is fixed and small, this negatively affected the efficiency of these methods.

In [1] the authors introduced performance comparisons between the standard methods of data cryptography and it was shown that the throughput of these methods ranges from 2683 to 10176 bytes per second, this throughput is very low, and using this method to encrypt-decrypt text files with big sizes will require a lot of time.

Some other authors introduced methods which enhanced the throughput, in [13] the authors introduced a method , the throughput was enhanced to reach 169.1 K bytes per second, while in [14] the introduced method enhanced the throughput to reach 710 K bytes per second.

In [2] the authors made performance comparisons between chaotic and non-chaotic methods of data cryptography, the final results of comparisons are shown in table 1.

Table 1: Cryptography methods performance comparisons [2]

Method	Throughput(K bytes per second)
Non-chaotic approach	170.3906
Chaotic approach	141.2305
Hyper Chaotic approach	636.3379

In [3] the authors provided a robust and fast image encryption scheme based on a mixing technique. In [4] the authors provided cosine-transform-based chaotic system for image encryption, while in [5] the authors introduced a novel image encryption algorithm based on polynomial combination of chaotic maps and dynamic function generation. In [6] the authors introduced Multiple-image Encryption Algorithm Based on DNA Encoding and Chaotic System, while in [7] the authors produced a multiple-image encryption with bit-plane decomposition and chaotic maps, these methods provided good quality a have various speeds as shown in table 2

Table 2: Performance comparisons of method mentioned in [3-7]

Method	Throughput(K bytes per second)
In [3]	888.8867
In [4]	638.4082
In [5]	911.0352
In [6]	360.4102
In [7]	384.9609

The Proposed Method

The proposed method uses a complex complicated PK to apply text file encryption-decryption, and it contains 12 components each has a double data type (64 bits length), the structure of PK is shown in figure 2.

PK											
Key 1				Key 2				Key 3			
R1	C1	r1	x1	R2	C2	r2	x2	R3	C3	r3	x3
Example											
30	40	3.99	0.01	100	25	3.97	0.011	70	80	3.88	0.009

Figure 2: PK structure

PK is divided into 3 parts, each part is to be used to generate a 2D chaotic logistic map model (CLMM) key [15], R and C are to be used to determine the key dimensions, where r and x are the parameters of CLMM, these parameters are used to generate the value of the 2D key as shown in the following sequence of operations:

```
R=100;C=100; % 2D key dimensions
r1=3.99;x1=.015;%Chaotic logistic map parameters
for i=1:R
   for j=1:C
      x1=r1*x1*(1-x1); %CLMM equation
      k11(i,j)=x1;
   end
end
k1=uint8(255*k11);
```

The generated key must be changed to integer and must be resized to match the text file dimensions.
The generated chaotic key are very sensitive to any minor changes in one or more components, this will be explained later in the implementation and results part.

Keys generation require time, this time will increase when increasing the key dimensions (see figure 3), so we can use a small key to minimize the generation time, this key can be resized to match the text file size, table 3 shows the required generation times for keys with various dimensions

Table 3: required keys generation times

R	C	size	Generation time(second)
4	5	20	0.000282
10	15	150	0.000297
20	20	400	0.000322

30	40	1200	0.000715
50	50	2500	0.001258
80	80	6400	0.000997
100	100	10000	0.001414
200	200	40000	0.006953
500	500	250000	0.167031
600	800	480000	0.423425
1000	1000	1000000	1.949329
1000	5000	5000000	11.427580

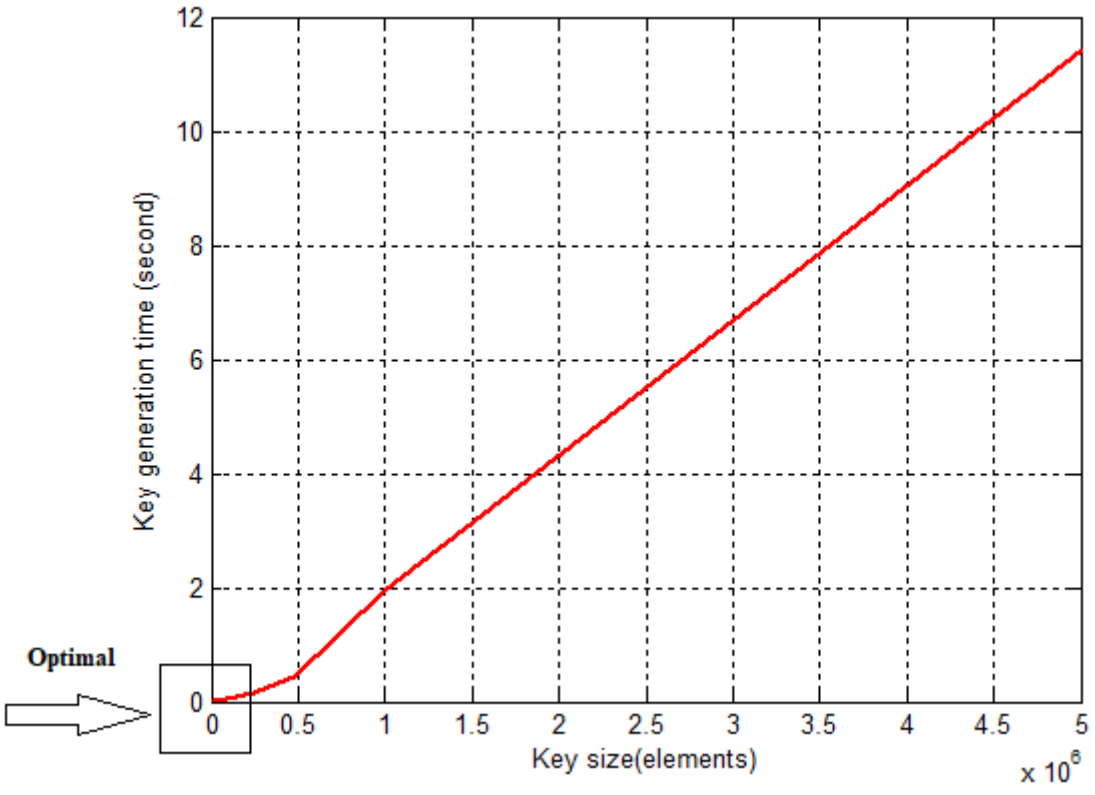

Figure 3: Key generation time vs key size

The encryption, decryption phases of the proposed method require 3 keys to apply 3 rounds of encryption decryption, the encryption phase is shown in figure 4 and it can be implemented applying the following algorithm:

Encryption phase:

Inputs

Text file to be encrypted, PK (the values of the 12 components shown in figure 2)

Output

Encrypted text file

Process

1) *Get the text file.*
2) *Retrieve the text file dimensions*
3) *Convert the text file to decimal*
4) *For each key do the following:*
    a) *Use the associated R and C to set the loops.*
    b) *Use the associated r and x to run the CLMM.*
    c) *Convert the key to integers between 0 and 255.*
    d) *Resize each key to match the text file size.*
5) *Apply first round of Xoring the text file with key1.*
6) *Apply second round of XORing the result of step 4 with key2.*
7) *Apply third round of XORing the result of step 5 with key3 to get the encrypted text file.*
8) *Convert the encrypted file from decimal to character.*

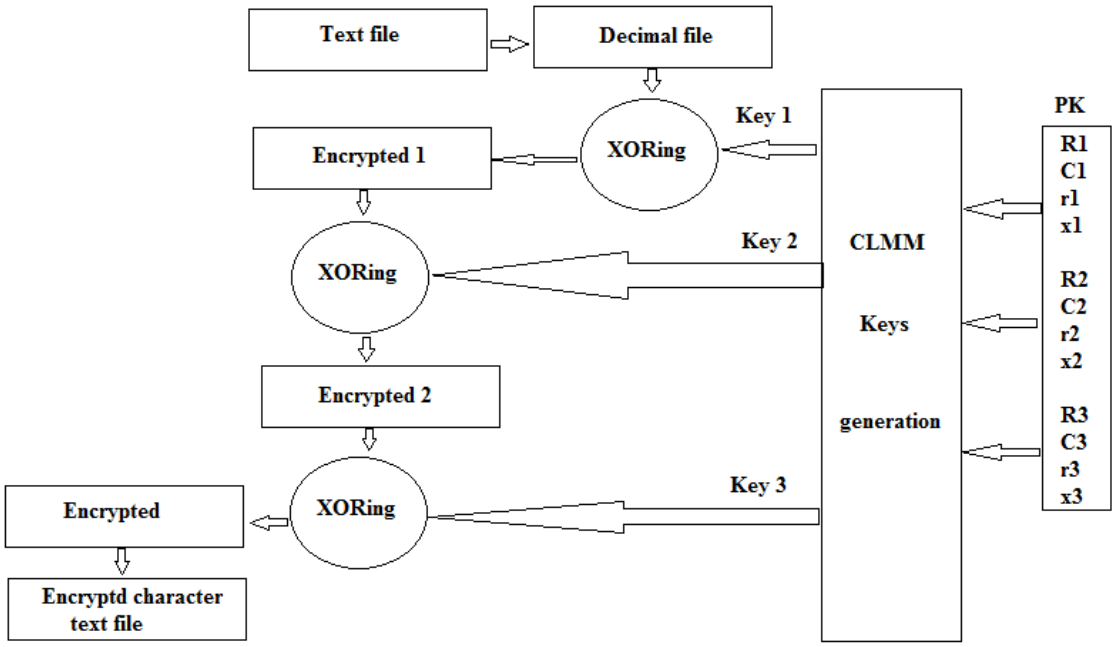

Figure 4: Proposed encryption phase

The decryption phase as shown in figure 5 can be implemented applying the following algorithm

Decryption phase

Inputs

Encrypted text file to be encrypted, PK (the values of the 12 components shown in figure 2)

Output

Decrypted text file

Process

1) *Get the encrypted text file.*
2) *Retrieve the text file dimensions*
3) *Convert the text file to decimal*
4) *For each key do the following:*
    a. *Use the associated R and C to set the loops.*
    b. *Use the associated r and x to run the CLMM.*
    c. *Convert the key to integers between 0 and 255.*
    d. *Resize each key to match the text file size.*

5) *Apply first round of Xoring the text file with key3.*

6) *Apply second round of XORing the result of step 4 with key2.*

7) *Apply third round of XORing the result of step 5 with key1to get the decrypted text file.*

8) *Convert the decrypted file from decimal to character.*

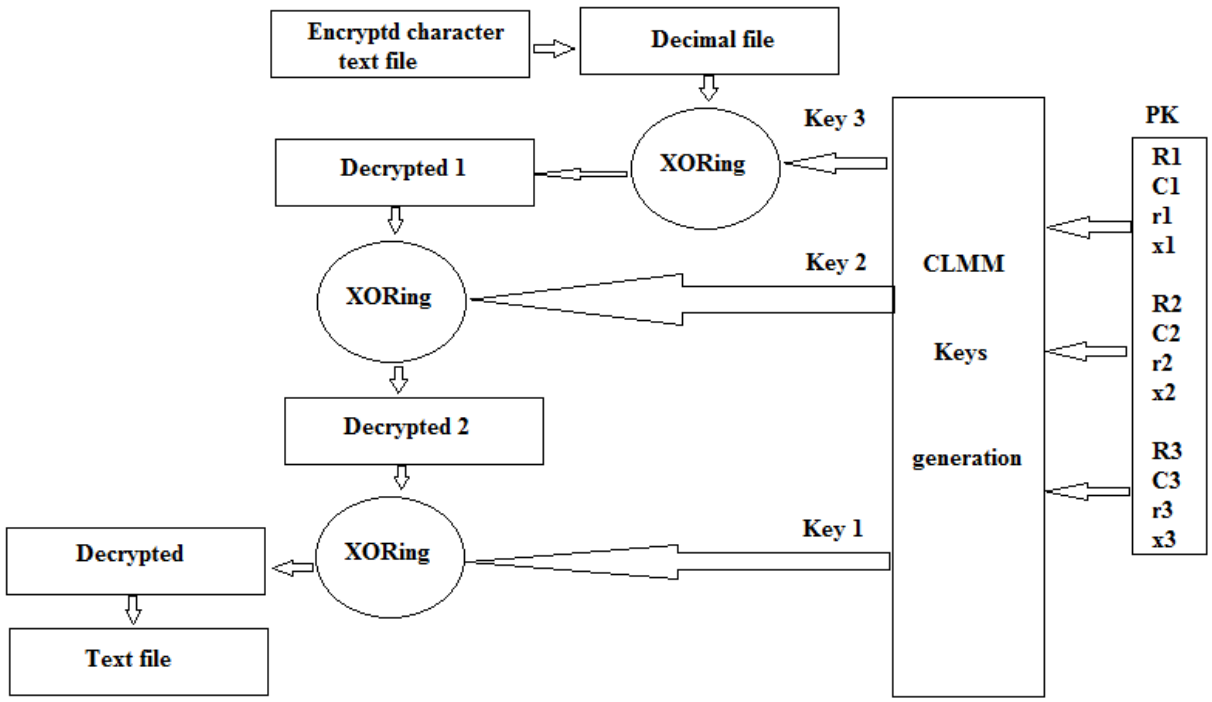

Figure 5: Proposed decryption phase

Implementation and Results Analysis

For researchers who want to reproduce the output, the following matlab code can be used.

```
%Text file information
a1='Text file cryptography';
a=uint8(a1);

[n1 n2]=size(a);
%First key generation
R=100;C=100;
r1=3.99;x1=.01;
for i=1:R
   for j=1:C
      x1=r1*x1*(1-x1);
      k11(i,j)=x1;
   end
end
k1=uint8(255*k11);
```

```
%Second key generation
R=120;C=130;
r1=3.99;x1=.011;
for i=1:R
   for j=1:C
      x1=r1*x1*(1-x1);
      k22(i,j)=x1;
   end
end
k2=uint8(255*k22);
%Third key generation
R=40;C=50;
r1=3.99;x1=.009;
for i=1:R
   for j=1:C
      x1=r1*x1*(1-x1);
      k33(i,j)=x1;
   end
end
k3=uint8(255*k33);
```

```
%Keys resizing
key1=imresize(k1,[n1 n2]);
key2=imresize(k2,[n1 n2]);
key3=imresize(k3,[n1 n2]);
%Encryotion
en1=bitxor(a,key1);
en2=bitxor(en1,key2);
en=bitxor(en2,key3);
%Output
corr2(a,en)
[m1 m2]=PSNR_RGB(a,en)
```

```
function [mse psnr]=PSNR_RGB(X,Y)
 X=double(X);
 Y=double(Y);

d1=max(X(:));
d2=max(Y(:));
d=max(d1,d2);
mse=mean2((X-Y).^2);

psnr=10*log((d.^2)./mse);
```

The decryption code is the same as encryption code using en as an input.

The proposed method was implemented using various images with various PK, below we will analyze the obtained results:

1) Sensitivity Analysis

PK contains 12 components divided into three parts, each part is to be used to generate one of the three keys used in the encryption-decryption phases. Each key is very sensitive to any minor changes in one or more PK components. The same components values must be used in the encryption and decryption phases. Any minor change in these values during the decryption phase will be considered as a hacking attempt resulting in producing a damaged corrupted decrypted text file, figures 6 and 7 show how the generated keys are sensitive to any changes in the PK components.

PK											
5	5	3.99	0.01	5	5	3.99	0.011	5	5	3.99	0.0091

Key1

10	39	131	254	3
13	47	154	243	45
147	248	27	95	238
64	191	190	193	188
197	178	215	136	253

r=3.99, x=0.01

Key2

11	42	141	252	13
50	160	238	63	190
194	185	202	166	231
88	230	91	233	80
219	124	254	4	14

r=3.99, x=0.011

Key3

9	35	121	254	5
19	71	204	163	235
73	208	153	244	41
138	253	9	34	119
253	7	27	97	240

r=3.99, x=0.0091

Figure 6: Different CLMM produce different keys

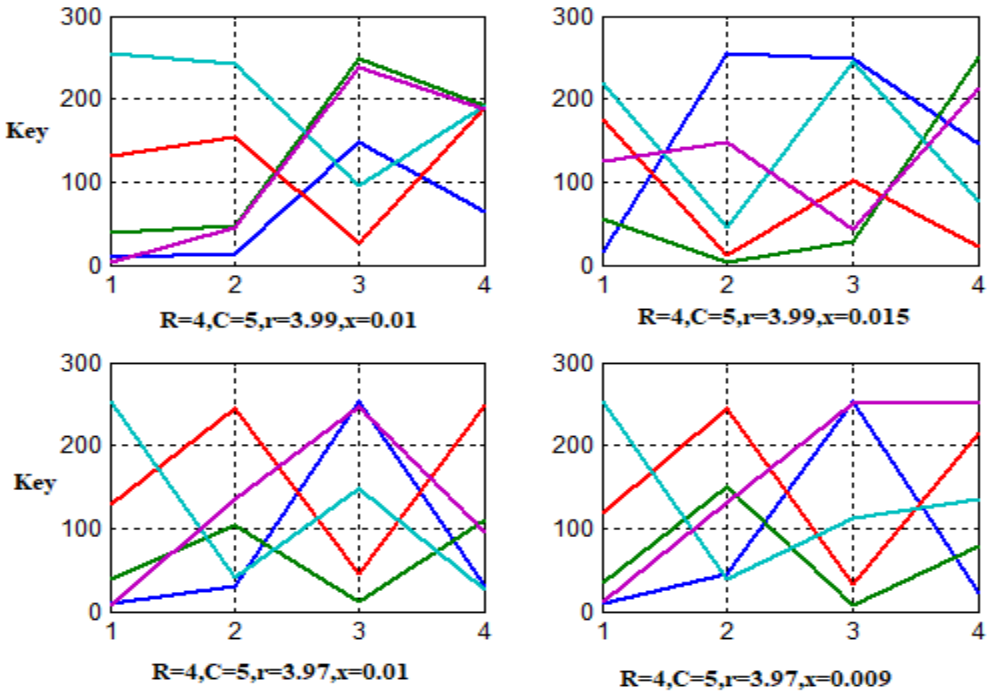

Figure 7: Keys sensitivity

Table 4 shows how changing PK components in the decryption phase will affect the decrypted text file

PK: Key used in the encryption phase

100,100,3.99,0.01,120,130,3.99,0.011,40,50,3.99,0.009

Text file: "Text file cryptography"

Table 4: Decrypted text file quality

Decryption key	Decrypted file	MSE	PSNR	CC
100,100,3.99,0.01,120,130,3.99,0.011,40,50,3.99,0.009	Text file cryptography	0	infinite	1
120,100,3.99,0.01,120,130,3.99,0.011,40,50,3.99,0.009	Vgzv"addc!buzr • mdz` • hp	86.3182	54.6765	0.9379
100,50,3.99,0.01,120,130,3.99,0.011,40,50,3.99,0.009	«cyq#een`$auyptography	348.5000	44.2969	0.7415
100,100,3.97,0.01,120,130,3.99,0.011,40,50,3.99,0.009	Plvp$jijb$`u□xxhkhjypy	30.5000	62.7065	0.9746
100,100,3.99,0.018,120,130,3.99,0.011,40,50,3.99,0.009	-el{'cj`l!hsyzzhe□l|jv	410.4091	42.8943	0.7217
100,100,3.99,0.01,120,130,3.99,0.019,40,50,3.99,0.009	ZckoÝ{ n 2z • ts • u qfhjk	2068.4	31.6179	-0.120
100,100,3.99,0.01,120,130,3.99,0.011,40,50,3.99,0.029	ª□□}Ín d • Ôxs□v}} fiwn	3478.4	25.5886	-0.86

2) Security Analysis

The PK contains 12 components, the R and C range is from 1 to 200 (to get a smaller key to minimize the key generation time, so 8 bits are required to represent each of the), r and x have a double type, so 64 bits are required to represent each of them.

The key space of the PK should be large enough to withstand attacks. In our proposed encryption-decryption algorithm, the initial values x, r, the varied parameters and the initial code book can be selected as secret keys. Let the largest precision be 2^{64} , the key space is about, R, and C values will be up to 1024 (10 bits) , while r and x have a double data type (64 bits)

$$\mathbf{Ks} = (2^{148})^3 = 2^{444}$$

Ks has a huge range it can be considered as a highly secure and hacking the PK will be A very difficult problem that takes time and effort to solve.

3) Quality Analysis

A method of text file cryptography is considered a good method if it destroys the data during the encryption phase and recover the original data in the decryption phase. MSE and PSNR can be used to measure the degrees of destruction and recovery. MSE between the input file and the encrypted file must be very high, while PSNR must be very low. MSE between the input file and the decrypted one must be zero, while PSNR must be infinite.

Some text files were selected and encrypted-decrypted using the proposed method, table 5 shows the calculated MSE and PSNR between the input text file and the encrypted file (R and C where equal 20) :

Table 5: Quality measurements

Text file size(K bytes)	Between input and encrypted files		Between input and decrypted files	
	MSE	PSNR	MSE	PSNR
9.7656	10855	17.9015	0	Infinite
15.2344	11006	17.7633	0	Infinite
21.9727	10887	17.8720	0	Infinite
39.0625	10712	18.0340	0	Infinite
244.1406	10875	17.8827	0	Infinite
976.5625	10853	17.9034	0	Infinite

From table 5 we can see that the obtained values for MSE and PSNR prove that the proposed method satisfied the quality requirements, changing the PK components did not negatively affect the quality.

4) Correlation Analysis

Several text files were selected and encrypted-decrypted using the proposed method, CC between the input file the encrypted file, and CC between the input file and decrypted file were calculated, table 6 shows the obtained results:

Table 6: CC results

Text file Size (K bytes)	CC between input and encrypted files	CC between input and decrypted files
9.7656	-0.0028	1
15.2344	-0.0021	1
21.9727	0.0040	1
39.0625	0.0016	1
244.1406	0.0016	1
976.5625	0.0020	1

The obtained results shown in table 6 prove that the proposed method totally destroyed the input file during the encryption phase (very low CC values), and total recover the original file in the decryption phase (CC always equal 1).

5) Speed Analysis

Method of text file cryptography must provide a high throughput, this can be achieved by minimizing the encryption and decryption times. In our proposed method the encryption time (decryption time) will depend on the size of the generated key, increasing this size as we saw earlier will increase the time required to generate the key, and thus the total encryption time will be increased.

The following PK with R=C=100 was selected and used to encrypt-decrypt several text files, table 7 shows the obtained results:

PK1:

100,100,3.99,0.01,120,130,3.99,0.011,40,50,3.99,0.009

Table 7: Times and throughputs using PK1

Text file size (K bytes)	Keys generation time(second)	Total encryption time (second)	Throughput (K bytes per second)
9.7656	0.0088	0.0368	265.7190
15.2344	0.0098	0.0480	317.3114
21.9727	0.0086	0.0494	445.1330
39.0625	0.0091	0.0486	803.2461
244.1406	0.0094	0.0617	1080.3
976.5625	0.0097	0.0929	10517
97656	0.0076	0.5113	10842

Increasing the keys dimension will increase the required times for cryptography, thus will decrease the method throughput, table 8 shows the obtained results of the proposed method using PK2

PK2:
1000,1000,3.99,0.01,1000,1000,3.99,0.011,1000,1000,3.99,0.009

Table 8: Times and throughputs using PK2

Text file size (K bytes)	Keys generation time(second)	Total encryption time (second)	Throughput (K bytes per second)
9.7656	6.2146	6.2871	1.5533
15.2344	6.0435	6.1165	2.4907
21.9727	6.2285	6.3047	3.4851
39.0625	6.1649	6.1649	6.2400
244.1406	6.5358	6.6342	36.8005
976.5625	5.8385	5.8929	165.7194

Decreasing the keys dimension will decrease the required times for cryptography, thus will increase the method throughput, table 9 shows the obtained results of the proposed method using PK3
PK3:

20,20,3.99,0.01,20,20,3.99,0.011,20,20,3.99,0.009

Table 9: Times and throughputs using PK3

Text file size (K bytes)	Keys generation time(second)	Total encryption time (second)	Throughput (K bytes per second)
9.7656	0.0052	0.0431	226.3915
15.2344	0.0045	0.0415	367.4085
21.9727	0.0050	0.0449	489.0126
39.0625	0.0053	0.0510	766.6830
244.1406	0.0050	0.0540	4518.9
976.5625	0.0040	0.0792	12334

From the previous tables we can see that selecting R and C with small values will maximize the throughput, and the method throughputs are better than those for the methods mentioned in the related work section.

6) Simplicity Analysis

The proposed method uses a simple sequence of operations which very easy to program. The proposed method can be used to encrypt-decrypt any text file with any size, using a selected PK. Changing the text file or the PK does not require any change in the sequence of operations.

Conclusion

A simple method of text files cryptography was introduced, this method can be used to encrypt-decrypt any text file with any size keeping the method highly efficient, changing the file and/or changing the PK does not require any changes in method algorithm. The proposed method was implemented using several text files with various sizes, the obtained results were analyzed, and the results of analysis showed that the proposed method satisfied the quality of cryptography by giving excellent results for MSE, PSNR and CC during both the encryption and decryption phases.

It was shown that the throughput provided by the proposed method is better than the throughput for other recently used method; the proposed method provided an increase in the throughput by minimizing the encryption and decryption times.

It was shown that the proposed method is very secure by using a complicated PK, the address space of the PK is very huge making the process of hacking impossible.

References

[1]. Aamer Nadeem, Dr M. Younus Javed, A Performance Comparison of Data Encryption Algorithms, Conference Paper · September 2005 DOI: 10.1109/ICICT.2005.1598556 · Source: IEEE Xplore.

[2]. M. Bala Kumara, P. Karthikkab , N. Dhivyac , T. Gopalakrishnan, A Performance Comparison of Encryption Algorithms for Digital Images, International Journal of Engineering Research & Technology (IJERT), Vol. 3 Issue 2, February – 2014.

[3]. Lee Mariel Heucheun Yepdia, Alain Tiedeu, and Guillaume Kom, A Robust and Fast Image Encryption Scheme Based on a Mixing Technique, Security and Communication Networks, Volume 2021 |Article ID 6615708 | https://doi.org/10.1155/2021/6615708

[4]. Z. Hua, Y. Zhou, and H. Huang, "Cosine-transform-based chaotic system for image encryption," *Information Sciences*, vol. 480, pp. 403–419, 2019.

[5]. M. Asgari-chenaghlu, M.-A. Balafar, and M.-R. Feizi-Derakhshi, "A novel image encryption algorithm based on polynomial combination of chaotic maps and dynamic function generation," *Signal Processing*, vol. 157, p. 1, 2019.

[6]. X. Zhang and X. Wang, *Multiple-image Encryption Algorithm Based on DNA Encoding and Chaotic System*, Springer, New York, NY, USA, 2019.

[7]. J. S. Zhenjun and R. Sun, "Multiple-image encryption with bit-plane decomposition and chaotic maps," *Optics and Lasers in Engineering*, vol. 80, pp. 1–11, 2016.

[8]. Diaa Salama Abdul. Elminaam, Hatem Mohamed Abdul Kader and Mohie Mohamed Hadhoud, "Performance Evaluation of Symmetric Encryption Algorithms", IJCSNS International Journal of Computer Science and Network Security, VOL.8 No.12, pp. 280-286, December 2008.

[9]. W. Stallings, Cryptography and Network Security, 4th Edition, Pearson Prentice Hall, 2006.

[10]. Singh S Preet, Mani Raman, "Comparison of Data Encryption Algorithms", International Journal of Computer science and Communications, Vol. 2, No.1, January-June 2011, pp. 125-127.

[11]. Singh Gurjeevan, Kumar Ashwani, Sandha K.S. "A Study of New Trends in Blowfish Algorithm" International Journal of Engineering Research and Applications (IJERA), Vol. 1, Issue 2, pp.321-326.

[12]. Agrawal Monika, Mishra Pradeep, "A Comparative Survey on Symmetric Key Encryption Techniques", International Journal on Computer Science and Engineering (IJCSE), Vol. 4 No. 05 May 2012, pp. 877-882.

[13]. Rania A. Elmanfaloty, Ehab Abou-Bakr, An Image Encryption Scheme Using a 1D Chaotic Double Section Skew Tent Map, Complexity, Volume 2020 |Article ID 7647421 | https://doi.org/10.1155/2020/7647421.

[14]. Vijayalakshmi, C., Lavanya, L., Navya, C.: A Hybrid Encryption Algorithm Based On AES and RSA. International Journal of Innovative Research in Computer and Communication Engineering 4(1), 909-917 (2016).

[15]. M. Atheer, "FPGA Based Image Encryption Using Chaotic Systems", PhD. thesis in Electronic Engineering University of Technology, October 2016.

[16]. Jamil Al-Azzeh, Bilal Zahran, Ziad Alqadi, Belal Ayyoub, Muhammed Mesleh: A Novel Based On Image Blocking Method to Encrypt-Decrypt Color JOIV: International Journal on Informatics Visualization, 2019.

[17]. Jamil Al-Azzeh, Bilal Zahran, Ziad Alqadi, Belal Ayyoub and Mazen Abu-Zaher: A Novel Zero-Error Method to Create a Secret Tag for an Image; Journal of Theoretical and Applied Information Technology 15th July 2018.

[18]. Qazem Jaber Rashad J. Rasras, Mohammed Abuzalata, Ziad Alqadi, Jamil Al-Azzeh; Comparative Analysis of Color Image Encryption-Decryption Methods Based on Matrix Manipulation: International Journal of Computer Science and Mobile Computing, Vol.8 Issue.2, 2019/3.

[19]. Rashad J. Rasras, Mohammed Abuzalata; Ziad Alqadi; Jamil Al-Azzeh; Qazem Jaber, Comparative Analysis of Color Image Encryption-Decryption Methods Based on Matrix Manipulation International Journal of Computer Science and Mobile Computing, Vol.8 Issue.3, March- 2019, pg. 14-26.

[20]. Dr. Mohamad Barakat Prof. Ziad Alqadi, Highly Secure Method for Secret Data Transmission, International Journal of Scientific Engineering and Science, vol. 6, issue 1, pp. 49-55, 2022.

[21]. Jihad Nadir, Ashraf Abu Ein, Ziad Alqadi, A Technique to Encrypt-decrypt Stereo Wave File, International Journal of Computer and Information Technology, vol. 5, issue 5, pp. 465-470, 2016.

[22]. Belal Zahran Rashad J Rasras, Ziad Alqadi, Mutaz Rasmi Abu Sara, B Zahran, Developing new Multilevel security algorithm for data encryption-decryption (MLS_ED), International Journal of Advanced Trends in Computer Science and Engineering, vol. 8, issue 6, pp. 3228-3235, 2019.

[23]. Ziad A AlQadi, Accurate Method for RGB Image Encryption, International Journal of Computer Science and Mobile Computing, vol. 9, issue 1, pp. 12-21, 2020.

[24]. Ziad Alqadi, Ahmad Sharadqh, Naseem Asad, Ismail Shayeb, Jamil Al-Azzeh, Belal Ayyoub, A highly secure method of secret message encoding, International Journal of Research in Advanced Engineering and Technology, vol. 5, issue 3, pp. 82-87, 2019.

[25]. Jamil Al-Azzeh, Ziad Alqadi, Qazem Jaber, A Simple, Accurate and Highly Secure Method to Encrypt-Decrypt Digital Images, JOIV: International Journal on Informatics Visualization, vol. 3, issue 3, pp. 262-265, 2019.

[26]. Ziad Alqad, Majid Oraiqat, Hisham Almujafet, Salah Al-Saleh, Hind Al Husban, Soubhi Al-Rimawi, A New Approach for Data Cryptography, International Journal of Computer Science and Mobile Computing, vol. 8, issue 9, pp. 30-48, 2019.

[27]. Aws AlQaisi, Mokhled AlTarawneh, Ziad A. Alqadi, Ahmad A. Sharadqah, Analysis of Color Image Features Extraction using Texture Methods, TELKOMNIKA, vol. 17, issue 3, pp. 1220-1225, 2019.

[28]. Abdullah I Alhasanat, Khaled D Matrouk, Haitham A Alasha'ary, Ziad A Al-Qadi, Connectivity-based data gathering with path-constrained mobile sink in wireless sensor networks, Wireless Sensor Network, Vol.6 No.6(2014), Article ID:47144,11 pages DOI:10.4236/wsn.2014.66013.

[29]. Musbah J Aqel, Ziad ALQadi, Ammar Ahmed Abdullah, RGB color image encryption-decryption using image segmentation and matrix multiplication, International Journal of Engineering & Technology, vol. 7, issue3.13, pp. 104-107, 2018.

[30]. Abdullah N Olimat, Ali F Al-Shawabkeh, Ziad A Al-Qadi, Nijad A Al-Najdawi, Ahmed Al-Salaymeh, Experimental study and computational approach prediction on thermal performance of eutectic salt inside a latent heat storage prototype, Thermal Science and Engineering Progress, Volume 37, 101606, 2023.

[31]. Mohammad S Khrisat, Ziad A Alqadi, Enhancing LSB Method Performance Using Secret Message Segmentation, IJCSNS, vol. 22, issue 7, pp. 1-6, 2022.

[32]. F. Omar and Z. A. Alqadi, M. M. Al-Laham, Digital Image Slicing to Strengthen the Security of LSB Technique of Encrypting Text Messages, 2022 International Arab Conference on Information Technology (ACIT), pp. 1-6, 2022.

[33]. M Khrisat, Ziad Alqadi, Performance evaluation of ANN models for prediction, Acadlore Trans. Mach. Learn, vol. 2, issue 1, pp. 13-20, 2023.

[34]. Ziad Alqadi, MOHAMAD-TARIQ BARAKAT, Speech Files Cryptography using Indices Keys and Blocking, International Journal of Computer Science and Mobile Computing, vol. 12, issue 3, pp. 22-33, 2023.

[35]. Ziad Alqadi Rashad J. Rasras , Mutaz Rasmi Abu Sara, Efficient Method to Message-Image Cryptography Using Reordered Image-Key, Traitement du Signal, vol. 40, issue 1, pp. 235-240, 2023.

[36]. Nasser Abdellatif Adnan Manasreh, Mohammad S. Khrisat, Hatim Ghazi Zaini, Ziad A. Alqadi, IMPROVED MEDIAN FILTER TO ELIMINATE SALT AND PEPPER NOISE IN DIGITAL, ARPN Journal of Engineering and Applied Science, vol. 18, issue 4, pp. 310-316, 2023.

[37]. Prof. Qazem Jabber Prof. Ziad Alqadi, Digital Image Cryptography using Index Keys, International Journal of Computer Science and Mobile Computing, vol. 12, issue 2, pp. 26 – 37, 2023.

[38]. Hatim Zaini, Ziad Alqadi, Improving Average and Median Filters, International Journal of Computer Science and Mobile Computing, vol. 12, issue 2, pp. 1-13, 2023.

[39]. Hatim Zaini, Ziad Alqadi, Long Messages Segmentation for Efficient Message Cryptography, International Journal of Computer Science and Mobile Computing, vol. 12, issue 2, pp. 59-69, 2023.

[40]. Ziad A. Alqadi and Nasser Abdellati Adnan Manasreh, Mohammad S. Khrisat, Hatim Ghazi Zaini, IMPROVING DATA STANDARD METHODS OF CRYPTOGRAPHY, ARPN Journal of Engineering and Applied Sciences, vol. 17, issue 24, pp. 2077-2088, 2023.

[41]. M. Abu-Faraj, A. Al-Hyari, C. Obimbo, K. Aldebei, I. Al-taharw Z. Alqadi, and O. Almanaseer, "Protecting Digital Image Protection Using Keys enhanced by 2D Chaotic Logistic Maps," Cryptography, vol. 7, Iss. 2, pp. 20-42, 2023, doi:10.3390/cryptography7020020.

[42]. M. Abu-Faraj, A. Al-Hyari, K. Aldebei, B. Al-Ahmad, and Z. Alqadi, "Rotation Left Digits to Enhance the Security Level of Message Blocks Cryptography," IEEE Access, vol. 10, pp. 69388- 69397, 2022, doi:10.1109/ACCESS.2022.3187317.

[43]. M. Abu-Faraj, A. Al-Hyari, I. Al-taharwa, B. Al-Ahmad, and Z. Alqadi, "CASDC: A Cryptographically Secure Data System Based on Two Private Key Images," IEEE Access, vol. 10, pp. 126304-126314, 2022, doi:10.1109/ACCESS.2022.32263

[44]. M. Abu-Faraj, A. Al-Hyari, and Z. Alqadi, "Experimental Analysis of Methods Used to Solve Linear Regression Models," CMC-Computers, Materials & Continua, vol. 72, no. 3, pp. 5699-5712, 2022, doi:10.32604/cmc.2022.027364.

[45]. M. Abu-Faraj, A. Al-Hyari, and Z. Alqadi, "Complex Matrix Private Key to Enhance the Security Level of Image Cryptography," Symmetry, vol. 14, Iss. 4, pp. 664-678, 2022, https://doi.org/10.3390/sym14040664.

[46]. M. Abu-Faraj, K. Aldebei, and Z. Alqadi, "Simple, Efficient, Highly Secure, and Multiple Purposed Method on Data Cryptography," Traitement du Signal, vol. 39, no. 1, pp. 173-178, 2022, doi:10.18280/ts.390117.

[47]. M. Abu-Faraj, and M. Zubi, "Analysis and Implementation of Kidney Stones Detection by Applying Segmentation Techniques on Computerized Tomography Scans," Italian Journal of Pure and Applied Mathematics, iss. 43, pp. 590-602, 2020.

[48]. M. Abu-Faraj, Z. Alqadi, B. Al-Ahmad, K. Aldebei, and B. Ali,"A Novel Approach to Extract ColorImage Features using Image Thinning," Applied Mathematics & Information Sciences (AMIS),vol.16, no. 5, pp. 665-672, 2022, doi:10.18576/amis/160501.

[49]. M. Abu-Faraj, A. Al-Hyari, B. Al-Ahmad, Z. Alqadi, B. Ali, and A.Alhaj, "Building a SecureImageCryptography System using Parallel Processing and Complicated Dynamic LengthPrivateKey," Applied Mathematics & Information Sciences (AMIS), vol. 16, no. 6, pp. 1017-1026, 2022,doi:10.18576/amis/160619

[50]. M. Abu-Faraj, A. Al-Hyari, I. Al-taharwa, Z. Alqadi, and B. Ali, "Increasing the Security ofTransmitted Text Messages Using Chaotic Key and Image Key Cryptography," InternationalJournal of Data and Network Science, vol. 7, no. 1, pp. 1-12, 2023, doi:10.5267/j.ijdns.2023.1.008.

[51]. M. Abu-Faraj, A. Al-Hyari, B. Al-Ahmad, Z. Alqadi, B. Ali, and K. Aldebe , "Improving theEfficiency of Median Filters Using Two Rounds of Generated Windows Processing," AppliedMathematics & Information Sciences (AMIS), vol. 17, no. 1, pp. 187-200, 2023, doi:10.18576/amis/170119.

[52]. M. Abu-Faraj, A. Al-Hyari, C. Obimbo, K. Aldebei, I. Al-taharw Z. Alqadi, and O. Almanaseer, "Protecting Digital Image Protection Using Keys enhanced by 2D Chaotic Logistic Maps," Cryptography, vol. 7, Iss. 2, pp. 20-24, 2023, doi:10.3390/cryptography7020020.

[53]. M. Abu-Faraj, Z. Alqadi, and M. Zubi, "Creating Color Image Features Based on Morphology Image Processing," Traitement du Signal, vol. 39, no. 3, pp. 797-803, 2022, doi:10.18280/ts.390304.

