Design of Encryption with Covertext and Reordering

by WT Handoko

IJITEB
Vol. 4
No. 2
2023

International Journal of Information Technology and Business

http://ejournal.uksw.edu/ijiteb

New Design of Encryption with Covertext and Reordering

Eka Ardhianto[a], Widiyanto Tri Handoko[a], Hari Murti[a], Rara Sri Artati Redjeki[a]

[a] Faculty of Information Technology and Industry, Universitas Stikubank, Semarang, Indonesia

Abstract: Documents for some entities are confidential and important, so security is required. Encryption with Covertext and Reordering (ECR) is a text-based document security model. ECR uses a random key to generate the ciphertext. The ECR random key is selected using a man-made method. This study aims to increase the level of document security based on the ECR mechanism. This paper proposes a new method using random keys in the permutation table. A random key is generated automatically by a function. Entropy is used as a measure of the security level of an encrypted document. Experiments show that the permutation table in the ECR gives better entropy values. This implies a better level of security. This experiment shows the entropy of the proposed method is 6.45 with an achievement of 97.58%. Thus, this ECR model will strengthen documents from intruders. The use of permutation tables also makes it easier to use ECR to secure documents.

Keywords: random, random key, permutation table, data hiding, information hiding

1. Introduction

Documents are important articles that contain information. The document shows the textual record [1]. Today's documents tend to be in digital form. Digital document representation is a textual document stored in bit strings [2]. Some entities consider documents as important and confidential assets. So that a document security mechanism is needed to maintain its confidentiality. One of the sciences that aim to secure documents is called cryptography [3]. Cryptography scrambles the information using a key so that third parties cannot access the information without the key [4]. Cryptography aims to secure documents by making them difficult to interpret [3] [5].

One of the cipher models is ECR (Encryption with Covertext and Reordering). ECR presents a combined text-based steganography approach that works on encryption techniques using XOR and reordering processes using Random Key [6]. The XOR approach provides the advantage of speeding up the encryption-decryption process [6]. The important parameters in ECR are covertext and random key. Covertext in the ECR as a key in the encryption-decryption process. The random key is used to combine enciphered text with covertext into the final ciphertext. Random keys in ECR are human generated. The random key contains four "1" and four "0". The placement of numbers in the random key is arranged in such a way that it is random.

Several studies have used random numbers for the encryption process. The use of Pseudorandom Number Generation (PRNG) to generate random values is applied to the document security process [7]. PRNG is also used to embed secret messages into covertext [8]. The random condition in the sudoku game is used as a solution in generating random values for the encryption process [9]. Random numbers using a Linear Congruential Generator (LGC) are also applied to embed characters into image pixels [10]. Random numbers are also generated using the position of the characters on the phonetic keyboard in Bengal letters [11].

In ECR, the random key consists of the numbers 1 and 0. The random key is preferably random. Random keys chosen by humans, will be vulnerable. Key selection by humans will tend to use keys related to personal data such as date of birth, name, telephone number, and address. If the personal data is known, it is possible that the key used will also be known. Humans are also more likely to use short, easy-to-remember keys for different processes.

This becomes difficult when the randomness of the Random key must be determined by humans. Random keys chosen by humans will be vulnerable. Key selection by humans will tend to use keys related to personal data such as date of birth, name, telephone number, and address. If the personal data is known, it is possible that the key used will also be known. Humans are also more likely to use short, easy-to-remember keys for different processes.

In this study, an approach using a random key permutation table and a random function is proposed to generate random conditions. This research changes the way the old key is issued using the selected key that is human generated to machine generated. The method used is to combine random techniques as a covertext generator and use random to select the reordering key in the permutation table. This new mechanism will have an impact on increasing the level of security of documents that are secured by more than 30% compared to the old model. The entropy value is also calculated and used as a comparison.

This article is written in several parts, the introduction in part 1, part 2 contains the proposed method, results, and discussion in part 3, and the last part contains conclusions.

2. Proposed Method

This section describes the proposed modification of the ECR model. The first version of the ECR process begins by dividing the plaintext into blocks with a size of 4 characters per block. The next step is determining the random key and the cover text for each block. The encryption process is performed by the XOR operator between the plaintext and cover text for each block. The reordering process is a process for creating the final ciphertext based on the determined random key value between enciphered text and covertext. Figure 1 shows the first version of ECR model.

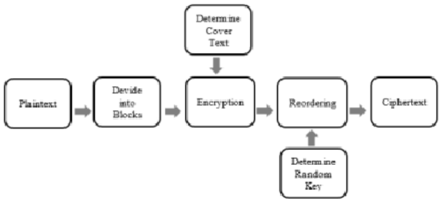

Figure 1. ECR model

The proposed modification is in the process of determining the cover text and the random keys for each block. In the first ECR version, the cover text and random keys were human-generated. This study uses a random permutation table to determine both. The cover text is generated using a random function between 32 and 256. This number represents the decimal number of printable characters. The encryption process is performed according to the ECR rules. A random key is generated using the random function on the permutation table index number. The obtained random numbers are converted into random keys according to the table. The reordering process is performed to produce the final ciphertext by the ECR rules. Figure 2 shows the proposed modification of ECR.

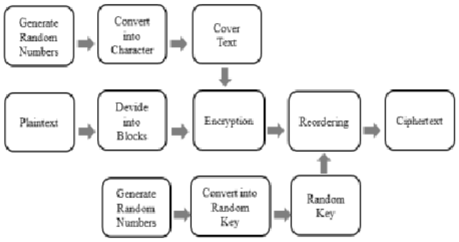

Figure 2. Modified ECR process

2.1. Proposed Permutation Table

The random key in ECR is used for reordering. The random key is used as a reference in preparing the final ciphertext from covertext and enciphered text. The random key consists of the numbers "0" and "1". The random key is 8 digits long. To compile a random key, 4 numbers of "0" and "1" are needed, each randomly placed. In this study, a random key is generated in a table with 2 parameters: the index as the serial number and the random key which represents the random key used. The number of random key permutations in the table uses equation 1.

$$P = n! / (k1!.k2!) \qquad (1)$$

The length of the random key is expressed as *n*. The number of digits 1 and digit 2 is expressed as *k1* and *k2*. Using equation 1, the permutation value (P) obtained 70 of random key arrangements. Table 1 shows the random key permutation table. This permutation table contains the list of keys obtained from the rules k1 and k2 based on the number of digits 1 and 0 allowed in the key array.

Table 1. Random key Permutation Table

Index	Decimal	Random key							
1	15	0	0	0	0	1	1	1	1
2	23	0	0	0	1	0	1	1	1
3	27	0	0	0	1	1	0	1	1
4	29	0	0	0	1	1	1	0	1
5	30	0	0	0	1	1	1	1	0
6	39	0	0	1	0	0	1	1	1
7	43	0	0	1	0	1	0	1	1
8	45	0	0	1	0	1	1	0	1
9	46	0	0	1	0	1	1	1	0
10	51	0	0	1	1	0	0	1	1
11	53	0	0	1	1	0	1	0	1
12	54	0	0	1	1	0	1	1	0

13	57	0	0	1	1	1	0	0	1
14	58	0	0	1	1	1	0	1	0
15	60	0	0	1	1	1	1	0	0
16	71	0	1	0	0	0	1	1	1
17	75	0	1	0	0	1	0	1	1
18	77	0	1	0	0	1	1	0	1
19	78	0	1	0	0	1	1	1	0
20	83	0	1	0	1	0	0	1	1
21	85	0	1	0	1	0	1	0	1
22	86	0	1	0	1	0	1	1	0
23	89	0	1	0	1	1	0	0	1
24	90	0	1	0	1	1	0	1	0
25	92	0	1	0	1	1	1	0	0
26	99	0	1	1	0	0	0	1	1
27	101	0	1	1	0	0	1	0	1
28	102	0	1	1	0	0	1	1	0
29	105	0	1	1	0	1	0	0	1
30	106	0	1	1	0	1	0	1	0
31	108	0	1	1	0	1	1	0	0
32	113	0	1	1	1	0	0	0	1
33	114	0	1	1	1	0	0	1	0
34	116	0	1	1	1	0	1	0	0
35	120	0	1	1	1	1	0	0	0
36	135	1	0	0	0	0	1	1	1
37	139	1	0	0	0	1	0	1	1
38	141	1	0	0	0	1	1	0	1
39	142	1	0	0	0	1	1	1	0
40	147	1	0	0	1	0	0	1	1
41	149	1	0	0	1	0	1	0	1
42	150	1	0	0	1	0	1	1	0
43	153	1	0	0	1	1	0	0	1
44	154	1	0	0	1	1	0	1	0
45	156	1	0	0	1	1	1	0	0
46	163	1	0	1	0	0	0	1	1
47	165	1	0	1	0	0	1	0	1
48	166	1	0	1	0	0	1	1	0
49	169	1	0	1	0	1	0	0	1
50	170	1	0	1	0	1	0	1	0
51	172	1	0	1	0	1	1	0	0
52	177	1	0	1	1	0	0	0	1
53	178	1	0	1	1	0	0	1	0
54	180	1	0	1	1	0	1	0	0
55	184	1	0	1	1	1	0	0	0
56	195	1	1	0	0	0	0	1	1
57	197	1	1	0	0	0	1	0	1
58	198	1	1	0	0	0	1	1	0
59	201	1	1	0	0	1	0	0	1
60	202	1	1	0	0	1	0	1	0
61	204	1	1	0	0	1	1	0	0
62	209	1	1	0	1	0	0	0	1
63	210	1	1	0	1	0	0	1	0
64	212	1	1	0	1	0	1	0	0
65	216	1	1	0	1	1	0	0	0
66	225	1	1	1	0	0	0	0	1
67	226	1	1	1	0	0	0	1	0
68	228	1	1	1	0	0	1	0	0
69	232	1	1	1	0	1	0	0	0
70	240	1	1	1	1	0	0	0	0

3. Result and Discussion

The difference between our design and the old one lies in the method of determining the covertext and randomkey. In the old design, the covertext and randomkey determination process was carried out by human generated. The covertext is a character and a random key consists of 8 digits binary number consisting of 4 digits 0 and 1 which are compiled manually by the sender. This will make the sender saturated if the number of plaintext blocks is large. This is what we see as a weakness in the ECR.

This research experiment uses data from https://haveibeenpwned.com/Passwords. The used data is the password character which is divided into 291 blocks. The first experiment performed data security using the same cover text and random key for each block. The second experiment uses the random function to cover text and permutation tables on the random key. The ECR encryption-decryption process was tested repeatedly to prove that there was no change between plaintext before encryption and plaintext after decryption.

In the encryption process, the plaintext is converted into ciphertext. At this stage, the ciphertext character frequency was calculated. In the first experiment, the cover text and the random key proceed by human generated. In the first experiment, the character's frequency was calculated from the final ciphertext. Figure 2 shows that there are 5 characters with a large number among other ciphertext characters. This character represents 4 characters of the cover text and 1 character of the random key. This condition arises because of the effect of the use of cover text and random keys which are implemented identically for each block. These five characters can raise suspicion for cryptanalysis as the use of repeated keys. Figure 3 also shows that the character distribution in the ciphertext is lacking. It shows that there are many unused characters. Figure 2 also illustrates that the frequency of using characters in the encryption process is not balanced. This will be a gap for cryptanalysts to open encrypted documents.

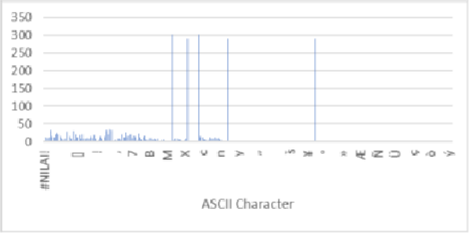

Figure 3. ECR ciphertext character histogram

In the second experiment, random functions and permutation tables are implemented. The cover text selection was performed randomly from 32 to 256 of a decimal value. It showed printable ASCII characters. The selected cover text is used as the second parameter for the encryption process. The random key is selected using the random function on the index number of the permutation table. The selected index number should be converted into a random key value according to the table. The selected random key is used as a parameter to establish the final ciphertext. The character frequency is calculated from the final ciphertext. The results obtained a better distribution of characters in the ciphertext. There is no use for characters that are too prominent. Figure 4 shows the distribution of characters in the ciphertext of this experiment. This random condition affected the use of characters more spread out in the ASCII character set. It means that almost all ASCII characters are used in the ciphertext. A better distribution of characters, it will make difficult for cryptanalysts to guess the actual contents of the secured document. This condition also makes the secured document will be hard to solve.

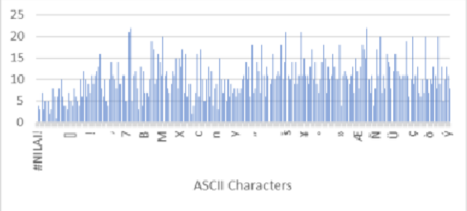

Figure 4. Modified ECR ciphertext character histogram

Table 2. The ciphertext entropy of ECR and Proposed Method.

Method	Entropy Value	Entropy Maximum	%
ECR [6]	4.34	6.61	65.66
Proposed Method	**6.45**	6.61	**97.58**

Entropy value is used to measure the information randomness [12]. The encryption product will be safer if it has a high entropy value. The higher the entropy value and the more ideal the entropy value, which will be more difficult to break the encryption system [13]. In this study, two ciphertexts will be compared to measure the entropy value. The first ciphertext is the ciphertext from the first experiment results which uses the same random key for all blocks. This first final ciphertext is stored as the ciphertext1 file. The second ciphertext resulted from the second experiment which implements the use of random functions and permutation tables of cover text and random key. This second final ciphertext is stored as ciphertext2 file. Calculation of the entropy value using cryptool. Table 2 shows the results of the entropy calculations for the two files.

The entropy value of the ciphertext from the second experiment showed a value of 6.45 from the maximum entropy value. It is better than the previous ciphertext which has 4.34 points of entropy value. There was an increase of more than 30%. This is affected by using the random function on the cover text and the random key used. The more random the ciphertext, the better the cipher model. So that the level of security in the second ciphertext is better.

4. Conclusion

From the results of the experiment and discussion, it is concluded that cryptography is a powerful and effective method for securing documents. The use of random functions on the model of cryptography algorithm will affect increasing the level of document security. The ECR which is combined with the permutation table and random function provide a higher entropy value, therefore it produces a better level of security than the previous version. Even though the entropy value has gotten better, another gap in the ECR may still exist. So other forms of ways to increase the level of security need to be considered continuously and become the focus of future research.

References

[1] Buckland M. K., What Is a "Document"?, JOURNAL OF THE AMERICAN SOCIETY FOR INFORMATION SCIENCE. 48(9) (1997), 804-809.

[2] Buckland M., What is a "digital document"?, Document Numérique. 2(2) (1998), 221-230.

[3] Ardhianto E, Trisetyarso A, Suparta W, Abbas B. S. and Kang C. H., Design Securing Online Payment Transactions Using Stegblock Through Network Layers, in INCITEST 2020. (2020), (Bandung).

[4] Babu V. S. and J H. K., A Study on Combined Cryptography and Steganography, International Journal of Research Studies in Computer Science and Engineering. 2(5) (2015), 45-49.

[5] Ardhianto E, Warnars H. L. H. S., Soewito B., Gaol F. L. and Abdurachman E, Improvement of Steganography Technique: A Survey, in 1st International Multidisciplinary Conference on Education, Technology, and Engineering (IMCETE 2019). (2020), (Serang).

[6] Kataria S, Singh K, Kumar T and Nehra M. S., ECR (Encryption with Cover Text and Reordering) based Text Steganography, in IEEE Second International Conference on Image Information Processing. (2013), (Waknaghat).

[7] Elmahi M. Y., Wahbi T. M. and Sayed M. H., Text Steganography Using Compression and Random Number Generators, International Journal of Computer Applications Technology and Research. 6(6) (2017), 259-263.

[8] Elmahi M. Y. and Wahbi T. M., Multi-Level Steganography Aided with Compression, in 2019 International Conference on Computer, Control, Electrical, and Electronics Engineering (ICCCEEE), (2019), (Khartoum).

[9] Majumder A, Changder S. and Debnath N. C., A New Text Steganography Method Based on Sudoku Puzzle Generation, in International Conference on Emerging Trends in Information Technology (ICETIT 2019), (2020), (New Delhi).

[10] Elveny M, Syah R., Jaya I and Affandi I., Implementation of Linear Congruential Generator (LCG) Algorithm, Most Significant Bit (MSB) and Fibonacci Code, in Compression and Security Messages Using Images, in 4th International Conference on Computing and Applied Informatics 2019 (ICCAI 2019), (2020), (Medan).

[11] Khairullah M., A novel steganography method using transliteration of Bengali text, Journal of King Saud University – Computer and Information Sciences 31(3) (2019), 348-366.

[12] Patil P, Narayankar P, G N. D. and M M. S., A comprehensive evaluation of cryptographic algorithms: DES, 3DES, AES, RSA and Blowfish, in International Conference on Information Security & Privacy (ICISP2015), (2016), (Nagpur).

[13] Rajesh S, Paul V, Menon V. G. and Khosravi M. R., A Secure and Efficient Lightweight Symmetric Encryption Scheme for Transfer of Text Files between Embedded IoT Devices, Symmetry 11(2) (2019), 1-21.

Design of Encryption with Covertext and Reordering

ORIGINALITY REPORT

5%	5%	0%	0%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	journal.upgris.ac.id Internet Source	3%
2	ejournal.uksw.edu Internet Source	2%

Exclude quotes	On	Exclude matches	Off
Exclude bibliography	On		