Tur6_Desain Baru Covertext dan Encryption Key Generator pada Model

by WT Handoko

Submission date: 28-Oct-2023 08:03AM (UTC+0700)
Submission ID: 2209575160
File name: esain_Baru_Covertext_dan_Encryption_Key_Generator_pada_Model.pdf (361.49K)
Word count: 2969
Character count: 14576

Jurnal Informatika dan Rekayasa Perangkat Lunak ISSN 2656-2855
Vol. 4, No. 2, September 2022, Hal. 92-97 e-ISSN 2685-5518

Desain Baru Covertext dan Encryption Key Generator pada Model Enkripsi ECR

Hari Murti[1*], Eka Ardhianto[2*], Indang Lestariningsih[3], Widiyanto Tri Handoko[4]
[1] Program Studi Sistem Informasi, Fakultas Teknologi Informasi dan Industri, Universitas Stikubank Semarang
[2,3,4] Program Studi Teknik Informatika, Fakultas Teknologi Informasi dan Industri, Universitas Stikubank Semarang
Jl. Tri Lomba Juang No. 1, Semarang 50241.
[*]Email: harimurti@edu.unisbank.ac.id, ekaardhianto@edu.unisbank.ac.id

Abstrak

Dokumen mungkin bersifat rahasia dan penting bagi entitas tertentu, sehingga diperlukan mekanisme keamanan khusus. Encryption with Covertext and Reordering (ECR) merupakan model keamanan dokumen berbasis teks. ECR memanfaatkan kunci random untuk memilih ciphertext. Kunci random dalam ECR diproses secara manual. Penelitian ini bertujuan untuk meningkatkan tingkat keamanan dokumen dengan menggunakan model enkripsi ECR. Tabel permutasi kunci acak yang diusulkan dalam model enkripsi ECR berubah. Nilai random diseleksi dengan otomatis menggunakan fungsi random pada tabel permutasi. Dalam penelitian ini salah satu ukuran yang digunakan adalah entropi, yaitu nilai tingkat keamanan dokumen terenkripsi. Hasil pengujian menunjukkan bahwa pengintegrasian tabel permutasi kunci acak memberikan nilai entropi yang lebih baik, yang menyiratkan tingkat keamanan yang lebih baik. Menggunakan tabel permutasi kunci acak juga memudahkan penggunaan ECR untuk mengamankan dokumen.

***Kata kunci**: kunci acak, pengacakan, pengamanan informasi, table permutasi*

PENDAHULUAN

Dokumen merupakan tulisan penting yang berisi informasi. Tulisan dalam dokumen secara umum berbentuk teks. Beberapa entitas memandang dokumen merupakan aset yang penting dan rahasia. Sehingga sebuah mekanisme pengamanan dokumen diperlukan untuk menjaga kerahasiaannya. Salah satu ilmu yang bertujuan untuk mengamankan dokumen dikenal sebagai kriptografi (Ardhianto et al., 2020). Kriptografi bertujuan mengamankan dokumen dengan cara membuat dokumen tersebut menjadi sulit untuk diartikan (Ardhianto et al., 2020) (Ardhianto, 2020). Kriptografi mengacak informasi menggunakan kunci sehingga pihak ketiga tidak dapat mengakses informasi tanpa kunci tersebut (Babu & J, 2015).

Salah satu model chiper adalah ECR (Encryption with Covertext and Reordering). ECR menyajikan pendekatan gabungan steganografi berbasis teks yang bekerja pada teknik enkripsi menggunakan Ex-OR dan proses menyusunan ulang menggunakan Random key (Kataria et al., 2013). Pendekatan penggunaan Ex-Or memberikan keuntungan dalam mempercepat proses enkripsi dekripsi (Kataria et al., 2013). Parameter penting dalam ECR adalah Covertext dan Random key. Covertext dalam ECR berguna sebagai kunci dalam proses enkripsi dekripsi. Random key digunakan untuk menggabungkan enciphertext dengan Covertext. Random key dibuat secara human generated. Random key berisi empat angka "1" dan empat angka "0". Penempatan angka dalam Random key disusun sedemikian rupa sehingga bersifat acak.

Penggunaan Pseudorandom Number Generation (PRNG) untuk membangkitkan nilai acak diaplikasikan pada proses pengamanan dokumen (Elmahi et al., 2017). PRNG juga digunakan untuk menanamkan pesan rahasia kedalam Covertext (Elmahi & M.Wahbi, 2019). Penggunaan kondisi acak pada permainan sudoku juga menjadi sebuah solusi dalam menggenerate nilai yang acak (Majumder et al., 2020). Penggunaan angka random menggunakan Linear Congruential Generator (LGC) juga diterapkan untuk penyematan karakter kedalam piksel gambar (Elveny et al., 2020). Angka random juga dihasilkan menggunakan posisi karakter pada fonetik keyboard dalam huruf benggali (Khairullah, 2019).

Dalam ECR, Random key tersusun atas angka 1 dan 0. Random key sebaiknya adalah

yang benar benar acak. Hal ini akan menjadi sulit ketika keacakan Random key harus ditentukan oleh manusia. Pada penelitian ini sebuah pendekatan penggunaan tabel permutasi Random key dan fungsi random diusulkan untuk menjadikan sebuah kondisi acak. Sebagai perbandingan hasil digunakan perhitungan entropy.

TINJAUAN PUSTAKA

Model enkripsi ECR bekerja pada menggunakan pendekatan teknik steganografi yang digabungkan dengan proses enkripsi berbasis Xor (Kataria et al., 2013). Operasi dari dua karakter dan menyusun ulang mereka yang akan lebih aman dan sulit untuk mengambil pesan asli dari teks terenkripsi. Model ini hadir menjawab permasalahan pada banyak metode Steganografi Teks yang ada adalah bahwa menggunakan teks sampul besar untuk menyembunyikan pesan rahasia dan juga membutuhkan terlalu banyak waktu dalam enkripsi dan dekripsi.

Proses enkripsi ECR melibatkan dua parameter penting yaitu Covertext dan Kunci. Teks terenkripsi disusun ulang menggunakan kunci acak 8-bit untuk menyembunyikan data kami dengan cara yang lebih aman. Kunci acak 8-bit berisi empat 1 dan empat 0 dimana bit 1 menggambarkan posisi teks terenkripsi kami dan bit 0 menggambarkan posisi covertext. Kunci digabungkan pada akhir untai dari teks terenkripsi untuk melakukan penyusunan ulang saat proses dekripsi. Gambar 1 dan Gambar 2 memperlihatkan proses enkripsi dan dekripsi pada ECR.

Proses pemilihan covertext dan kunci pada model ECR versi awal semua diserahkan oleh pengguna. Hal ini menyebabkan kerentanan terhadap informasi yang dirahasiakan. Hal ini disebabkan manusia lebih cenderung memilih sesuatu yang mudah dan bersifat berulang. Terlebih manusia akan memilih kunci yang dekat dengan ingatan serta data pribadinya. Kondisi ini dapat berulang dengan informasi yang berbeda akan menggunakan kunci yang sama.

Teks Pesan	a		b		c		d	
Tentukan Random Key	1	0	1	1	0	1	0	0
Random Key (dalam karakter)	10110100 = ╠							

Tentukan Covertext	F	-	N	e
Proses Enkripsi	a	b	c	d
	Xor	Xor	Xor	Xor
	F	-	N	e
Hasil Enkripsi	‘	O	-	d
Penyusunan Akhir (reordering)	‘ F O - - d N e			
Hasil Akhir (ditambah Random Key)	‘FO--dNe ╠			

Gambar 1. Proses Enkripsi ECR

Teks Akhir	‘FO--dNe ╠			
Uraikan Random Key	1 0 1 1 0 1 0 0			
Teks Rahasia	‘ F O - - d N e			
Covertext	F	-	N	e
Teks Rahasia	‘	O	-	d
Proses Dekripsi	‘	O	-	d
	Xor	Xor	Xor	Xor
	F	-	N	e
Teks Pesan	a	b	c	d

Gambar 2. Proses Dekripsi ECR

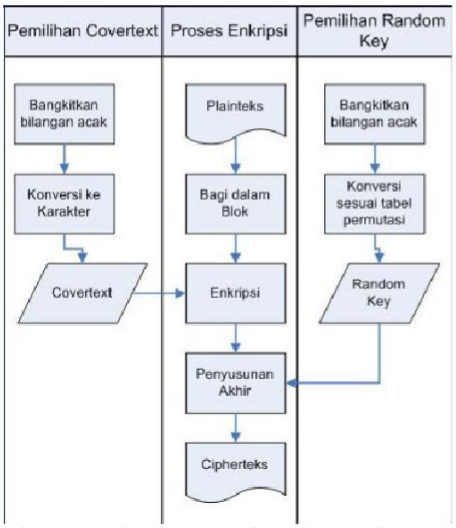
METODE

Pada bagian ini akan dijelaskan usulan modifikasi model ECR. Proses ECR diawali dengan membagi Plaintext menjadi beberapa blok dengan ukuran 4 karakter setiap blok. Langkah selanjutnya ialah menentukan Random key dan Covertext untuk setiap blok. Proses enkripsi dilakukan menggunakan operator x-or kepada Plaintext dan Covertext pada masing masing blok. Proses reordering ialah melakukan penyusunan ciphertext akhir berdasar nilai Random key antara enciphertext dan Covertext.

Bentuk modifikasi yang diusulkan ialah pada bagian penentuan Covertext dan Random key. Pada mulanya, pembangkitan dilakukan secara human generated. Penelitian ini menggunakan fungsi random dan penggunaan tabel permutasi untuk menentukan keduanya. Covertext ditentukan dengan diawali penentuan 4 angka acak antara 32 hingga 256. Angka tersebut ialah angka desimal yang menyatakan *printable character*. Proses enkrpsi tetap dilakukan sesuai aturan ECR. Random key ditentukan dengan menggunakan fungsi random pada angka indeks tabel permutasi. Angka random yang didapat dikonversikan menjadi Random key sesuai pada tabel. Proses reordering dilakukan sesuai dengan proses

ISSN 2656-2855
e-ISSN 2685-5518

ECR, hingga menghasilkan ciphertext akhir. Gambar 3 menunjukkan bagan ECR yang telah dimodifikasi.



Gambar 3. Usulan Modifikasi Model Enkripsi ECR

Random key dalam ECR berperan pada tahap *reordering*. *Random key* digunakan sebagai acuan dalam penyusunan *ciphertext* akhir dari *Covertext* dan *enciphertext*. *Random key* terdiri dari angka "0" dan "1". *Random key* memiliki panjang 8 angka. Untuk menyusun *Random key* diperlukan angka "0" dan "1" masing masing 4 buah yang disusun secara bebas. Dalam penelitian ini, *Random key* disusun pada tabel dengan 2 parameter yaitu: parameter indeks sebagai nomor urut dan parameter *Random key* yang menyatakan *Random key* yang digunakan. Penyusunan *Random key* pada tabel menggunakan persamaan 1.

$$P = \frac{n!}{k_1! \, x \, k_2!} \quad (1)$$

Panjang *Random key* dinyatakan sebagai n. Jumlah digit 1 dan digit 2 dinyatakan sebagai k1 dan k2. Menggunakan persamaan 1, maka nilai permutasi (P) yang didapatkan ialah 70 susunan *Random key*. Tabel 1 menunjukkan tabel permutasi *Random key* yang digunakan.

Tabel 1. Tabel Permutasi Random key

Indek	Angka Desimal	*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

ISSN 2656-2855
e-ISSN 2685-5518

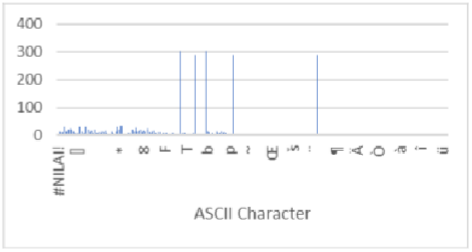
58	198	1	1	0	0	0	1	1	0
59	201	1	1	0	0	1	0	0	1
60	202	1	1	0	0	1	0	1	0
61	204	1	1	0	0	1	1	0	0
62	209	1	1	0	1	0	0	0	1
63	210	1	1	0	1	0	0	1	0
64	212	1	1	0	1	0	1	0	0
65	216	1	1	0	1	1	0	0	0
66	225	1	1	1	0	0	0	0	1
67	226	1	1	1	0	0	0	1	0
68	228	1	1	1	0	0	1	0	0
69	232	1	1	1	0	1	0	0	0
70	240	1	1	1	1	0	0	0	0

HASIL DAN PEMBAHASAN

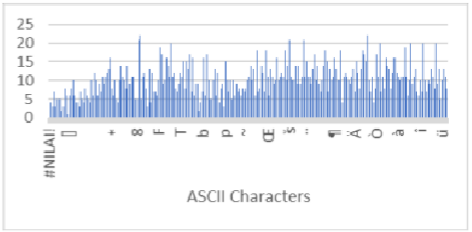
Ekperimen penelitian ini menggunakan data dari https://haveibeenpwned.com/ Passwords. Data yang digunakan ialah karakter password yang kemudian terbagi menjadi 291 blok. Eksperimen yang dilakukan ada dua macam. Eksperimen pertama melakukan pengamanan data menggunakan Covertext dan Random key yang sama untuk setiap blok. Ekperimen kedua menggunakan fungsi random untuk Covertext dan tabel permutasi pada Random key. Proses enkripsi dekripsi ECR dilakukan secara bolak balik untuk membuktikan bahwa tidak ada perubahan antara Plaintext sebelum di enkripsi dan Plaintext setelah didekripsi.

Pada proses enkripsi, Plaintext akan diubah menjadi ciphertext. Pada tahap ini dilakukan penghitungan frekuensi karakter yang membentuk ciphertext. Pada ekperimen pertama, Covertext dan Random key di tentukan secara human generated. Gambar 4 memperlihatkan bahwa terdapat 5 karakter dengan jumlah yang menonjol diantara karakter ciphertext lainnya. Karakter tersebut merupakan representasi dari 4 karakter Covertext dan 1 karakter Random key. Hal ini karena penggunaan Covertext dan Random key yang diimplementasikan secara identik untuk setiap blok. Gambar 4 juga menunjukkan bahwa distribusi karakter pada ciphertext tidak merata. Dapat dilihat bahwa terdapat karakter yang tidak digunakan. Hal ini akan menjadi celah bagi kriptanalis untuk membuka dokumen yang dienkripsikan.

Pada eksperimen kedua, penggunaan fungsi random dan tabel permutasi digunakan. Covertext dipilih secara random dari printable karakter ASCII. Random key dipilih dengan menggunakan fungsi random pada nomor indeks tabel permutasi, yang kemudian dikonveriskan pada angka Random key sesuai indek. Hasil yang didapatkan ialah bahwa sebaran karakter pada ciphertext lebih merata. Tidak ada penggunan karakter yang terlalu menonjol. Gambar 5 memperlihatkan sebaran karakter pada ciphertext dari eksperimen ini. Dengan sebaran karakter yang lebih merata, hal ini akan menjadikan kriptanalis menjadi lebih sulit untuk menebak isi dokumen sebenarnya.



Gambar 4. Histogram karakter cipherteks ECR versi asli



Gambar 5. Histogram karakter cipherteks ECR versi modifikasi

Entropi digunakan untuk mengukur keacakan dari sebuah informasi (Patil et al., 2016). Sebuah hasil enkripsi akan menjadi lebih aman jika memiliki nilai entropi yang lebih tinggi. Semakin tinggi nilai entropi dan semakin ideal nilai entropi, mana untuk membobol sistem enkripsi akan semakin sulit (Rajesh et al., 2019). Dalam penelitian ini akan membandingkan dua buah ciphertext yang akan diukur nilai entropinya. Ciphertext pertama ialah ciphertext dari hasil ECR yang menggunakan Random key yang sama untuk semua blok, ciphertext kedua ialah yang mengimplementasikan penggunan fungsi random dan tabel permutasi. perhitungan nilai entropi menggunakan perangkat cryptool. Tabel 2 memperlihatkan hasil perhitungan entropi kedua file.

Tabel 2. Perbandingan Entropi ECR versi asli dan ECR versi modifikasi

Metode	Nilai Entropi	Entropi Maksimum	%
ECR	4.34	8	54,25
ECR Modifikasi	6.45	8	80,62

Nilai entropi ciphertext hasil eksperimen kedua menunjukkan nilai 6.45 yang lebih baik dari ciphertext sebelumnya yaitu 4.34. Terjadi peningkatan lebih dari 25%. Hal ini merupakan efek dari penggunaan bentuk acak pada Covertext dan Random key yang digunakan. Semakin acak ciphertext maka akan semakin baik model ciphernya. Demikian pula dapat dikatakan bahwa tingkat keamanan pada ciphertext kedua adalah lebih baik.

KESIMPULAN

Berdasarkan hasil penelitian, dapat ditarik simpulan bahwa penggunaan bentuk random akan memberikan efek pada peningkatan level keamanan dokumen. ECR modifikasi yang diusulkan akan memberikan tingkat pengamanan yang lebih baik dari versi sebelumnya. Meski nilai entropi sudah menjadi lebih baik, celah celah lain dalam ECR mungkin masih ada. Sehingga bentuk cara lain untuk meningkatkan level keamanan perlu difikirkan secara kontinyu dan menjadi fokus penelitian kedepan.

DAFTAR PUSTAKA

Ardhianto, E. (2020). Improvement of Steganography Technique: A Survey. *1st International Multidisciplinary Conference on Education, Technology, and Engineering (IMCETE 2019)*, 289–292. www.scimagojr.com.

Ardhianto, E., Trisetyarso, A., Suparta, W., Abbas, B. S., & Kang, C. H. (2020). Design Securing Online Payment Transactions Using Stegblock through Network Layers. *IOP Conference Series: Materials Science and Engineering*, *879*(1). https://doi.org/10.1088/1757-899X/879/1/012027

Babu, V. S., & J, H. K. (2015). A Study on Combined Cryptography and Steganography. In *International Journal of Research Studies in Computer Science and Engineering (IJRSCSE)* (Vol. 2, Issue 5). www.arcjournals.org

Elmahi, M. Y., & M.Wahbi, T. (2019). Multi-Level Steganography Aided with Compression. *2019 International Conference on Computer, Control, Electrical, and Electronics Engineering (ICCCEEE)*, 1–6. https://doi.org/10.1109/ICCCEEE46830.2019.9071188

Elmahi, M. Y., Wahbi, T. M., & Sayed, M. H. (2017). Text Steganography Using Compression and Random Number Generators. In *International Journal of Computer Applications Technology and Research* (Vol. 6, Issue 6). www.ijcat.com

Elveny, M., Syah, R., Jaya, I., & Affandi, I. (2020). Implementation of Linear Congruential Generator (LCG) Algorithm, Most Significant Bit (MSB) and Fibonacci Code in Compression and Security Messages Using Images. *Journal of Physics: Conference Series*, *1566*(1). https://doi.org/10.1088/1742-6596/1566/1/012015

Kataria, S., Kumar, T., Singh, K., & Nehra, M. S. (2013). ECR (encryption with cover text and reordering) based text steganography. *2013 IEEE Second International Conference on Image Information Processing (ICIIP-2013)*, 612–616. https://doi.org/10.1109/ICIIP.2013.6707666

Khairullah, M. (2019). A novel steganography method using transliteration of Bengali text. *Journal of King Saud University - Computer and Information Sciences*, *31*(3), 348–366. https://doi.org/10.1016/j.jksuci.2018.01.008

Majumder, A., Changder, S., & Debnath, N. C. (2020). A New Text Steganography Method Based on Sudoku Puzzle Generation. In *Singh, P., Panigrahi, B., Suryadevara, N., Sharma, S., Singh, A. (eds) Proceedings of ICETIT 2019. Lecture Notes in Electrical Engineering* (Vol. 605, pp. 961–972). Springer. https://doi.org/10.1007/978-3-030-30577-2_85

Patil, P., Narayankar, P., Narayan D.G., & Meena S.M. (2016). A Comprehensive

Evaluation of Cryptographic Algorithms: DES, 3DES, AES, RSA and Blowfish. *Procedia Computer Science*, *78*, 617–624. https://doi.org/10.1016/j.procs.2016.02.108

Rajesh, S., Paul, V., Menon, V. G., & Khosravi, M. R. (2019). A secure and efficient lightweight symmetric encryption scheme for transfer of text files between embedded IoT devices. *Symmetry*, *11*(2). https://doi.org/10.3390/sym11020293

Tur6_Desain Baru Covertext dan Encryption Key Generator pada Model

ORIGINALITY REPORT

4%	4%	1%	1%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	publikasiilmiah.unwahas.ac.id Internet Source	2%
2	journal3.uin-alauddin.ac.id Internet Source	1%
3	Edy Winarno, Imam Husni Al Amin, Wiwien Hadikurniawati. "Asymmetrical Half-join Method on Dual Vision Face Recognition", International Journal of Electrical and Computer Engineering (IJECE), 2017 Publication	1%
4	www.publikasiilmiah.unwahas.ac.id Internet Source	<1%
5	download.garuda.ristekdikti.go.id Internet Source	<1%

Exclude quotes	On	Exclude matches	Off
Exclude bibliography	On		