IMPLEMENTASI KEAMANAN DATABASE MENGGUNAKAN KRIPTOGRAFI RC4 PADA SISTEM MILIK PT. TOROP SUMBER MAKMUR

Dadan Romadhan[1*], Ferdiansyah Ferdiansyah[2]

[1,2] Fakultas Teknologi Informasi, Teknik Informatika, Universitas Budi Luhur, Jakarta, Indonesia

Email: [1*]Dadanromadhan20@gmail.com, [2]ferdiyansyah@budiluhur.ac.id
(* : corresponding author)

Abstrak-PT. Torop Sumber Makmur adalah sebuah perusahaan yang bergerak di bidang penyediaan jasa pembersihan mesin yang membutuhkan penanganan khusus, selain itu mereka juga menyediakan bahan kimia untuk keperluan pembersihan alat atau cleaning agent. Meski pun perusahaan ini bergerak di bidang jasa pembersihan, PT. Torop Sumber Makmur juga memiliki data perusahaan yang sangat penting dan data tersebut seperti data customer yang berbahaya jika sampai tersebar luas dan di manfaatkan oleh oknum yang tidak bertanggung jawab. Sebelumnya, PT. Torop Sumber Makmur masih menggunakan metode konvensional dalam penangan dan penyimpanan data-datanya, yang tentu saja hal ini sangat berbahaya karena tidak bisa di jamin keamanannya. Di lansir dari situs medcom, menurut Pratama Persadha selaku chairman Lembaga Riset Keamanan Siber Indonesia *Communication and Information System Security Research Center* (CISSReC) mengatakan bahwa Indonesia masih belum memiliki aturan kemanan siber setingkat Undang-Undang(UU), salah satunya adalah RUU Perlindungan Data Pribadi (PDP) sehingga bisa di katakana regulasinya masih lemah. Maka dari itu, PT. Torop Sumber Makmur sangat membutuhkan sebuah aplikasi yang dapat menjamin kerahasiaan data perusahaan seperti data proyek dan data pelanggan. Dalam mengatasi masalah kemanan tersebut, maka di butuhkan suatu metode yang akan mempersulit oknum yang berhasil membobol data tersebut, dan salah satu caranya adalah menggunakan enkripsi. Metode atau algoritma yang akan di terapkan dalam penelitian ini adalah algoritma RC4 yang mempunyai sifat stream cipher. RC4 adalah algoritma berjenis simetris. Adapun kelemahan yang di miliki oleh algoritma simetris adalah proses enkripsi dan dekripsi yang menggunakan kunci yang sama. Untuk mengatasi masalah tersebut, di lakukan pengacakan kunci pada setiap record di database. Dengan cara tersebut, dapat meminimalisir kemungkinan terjadinya data leakage oleh oknum yang tidak bertanggung jawab. RC4 juga memiliki kelebihan, yaitu memiliki kecepatan dalam proses enkripi dan dekripsinya. Hasil yang di dapatkan dari penelitian ini yaitu isi record dalam database dalat di enkripsi dengan baik, serta proses enkripsi maupun deskripsi yang sangat cepat serta memiliki tingkat keamanan yang cukup mempuni. Hasil dari percobaan yang di lakukan, di ketahui rata-rata durasi dari sebuah proses enkripsi pada suatu record adalah 0.356941236 ms, 3025.2 byte dan rata-rata lama proses enkripsi dari suatu record adalah adalah 0.538033112 ms, 3025.2 byte.

Kata Kunci: Kriptografi, RC4 Stream Cipher, Enkripsi, Dekripsi.

IMPLEMENTATION OF DATABASE SECURITY USING RC4 CRYPTOGRAPHY IN THE SYSTEM OF PT. TOROP SUMBER MAKMUR

Abstract-*PT. Torop Sumber Makmur is a company engaged in providing cleaning services for machines that require special handling, besides that they also provide chemicals for cleaning tools or cleaning agents. Although this company is engaged in cleaning services, PT. Torop Sumber Makmur also has very important company data and that data is like customer data which is dangerous if it gets widespread and is used by irresponsible people. Previously, PT. Torop Sumber Makmur still uses conventional methods in handling and storing its data, which of course is very dangerous because security cannot be guaranteed. As reported from the medcom website, Pratama Persadha as chairman of the Indonesian Cybersecurity Research Institute Communication and Information System Security Research Center (CISSReC) said that Indonesia still does not have cyber security rules at the level of the Act (UU), one of which is the Personal Data Protection Bill. (PDP) so it can be said that the regulations are still weak. Therefore, PT. Torop Sumber Makmur desperately needs an application that can guarantee the confidentiality of company data such as project data and customer data. In overcoming these security problems, we need a method that will make it difficult for individuals who managed to break into the data, and one way is to use encryption. The method or algorithm that will be applied in this research is the RC4 algorithm which has stream cipher properties. RC4 is an algorithm of symmetric type. The weakness of the symmetric algorithm is that the encryption and decryption processes use the same key. To solve this problem, a randomization key is performed on each record in the database. In this way, it can minimize the possibility of data leakage by irresponsible persons. RC4 also has advantages, namely having speed in the encryption and decryption process. The results obtained from this study are that the contents of the records in the database can be encrypted properly, and the encryption and description processes are very fast and have a sufficient level of security. From the results of the experiments carried out, it is known that the average duration of an encryption process on a record is 0.356941236 ms, 3025.2 bytes and the average length of the encryption process of a record is 0.538033112 ms, 3025.2 bytes.*

Keywords*: Cryptography, RC4 Stream Cipher, Encryption, Decryption.*

1. PENDAHULUAN

Di kutip dari tekno.kompas.com, sepanjang tahun 2020 muncul beberapa kasus kebocoran data yang di alami pemerintah ataupun perusahaan swasta seperti salah satu contohnya adalah
Platform e-commerce dan salah satu dari perusahaan tersebut adalah Tokopedia yang pada
awal bulan Mei 2020 di kabarkan menjual data pengguna sebanyak 91 juta data dan lebih dari tujuh juta data merchant Tokopedia yang bersifat privasi seperti gender, lokasi, *username*, nama lengkap, alamat email, nomor ponsel pribadi, dan *password*.

Dari berita tersebut menjelaskan bahwa data adalah suatu *asset* yang sangat berharga, dan terlebih lagi jika data tersebut di salah guna kan maka pemilik data tersebut lah yang akan paling merasakan akibatnya. Salah satu cara menghindari kebocoran data ini adalah dengan teknik kriptografi, kriptografi secara definisi umum adalah *kryptos* dan *graphein* dalam bahasa Yunani. Masing-masing mempunyai arti yang berbeda, *kryptos* mempunyai arti rahasia atau tersembunyi, sedangkan *graphein* mempunyai arti menulis. Jadi secara umum, kriptografi adalah proses menulis atau menyampaikan pesan secara rahasia.
Sedangkan dalam teknologi informasi, kriptografi adalah suatu teknik enkripsi dimana data asli (*plaintext*) kemudian akan diacak menggunakan suatu kunci enkripsi sehingga akan sangat sulit untuk di baca (*ciphertext*). Kemudian, *database* adalah kumpulan data atau informasi yang di simpan secara sistematis. *Database* ini berfungsi sebagai penyimpanan data operasional, pelanggan, data kontrak, maupun data yang bersifat privasi lainnya dari suatu perusahaan atau pun instansi.

PT. Torop Sumber Makmur adalah perusahaan yang menawarkan berbagai jasa dan juga menjual bahan-bahan kimia untuk kebutuhan *maintenance* dari alat-alat yang butuh penanganan khusus dan professional. Mereka memiliki pelanggan yang cukup besar dan jika data tersebut sampai tersebar luas dan juga di salah gunakan oleh oknum yang tidak bertanggung jawab, maka kedua belah pihak akan merasa sangat di rugikan. Selama ini, proses penyimpanan data yang mereka miliki dalam mengelola proyek-proyek masih di lakukan secara konvensional dan tanpa adanya pengamanan yang cukup menjajikan. Kemudian data tersebut juga tidak boleh di akses oleh sembarang orang, dan juga sudah seharusnya menggunakan cara yang lebih efisien serta pengamanan yang baik dalam menyimpan data tersebut. Metode kriptografi yang akan di gunakan adalah algoritma kriptografi RC4 yang mempunyai sifat *stream cipher*, sehingga data yang sudah di enkripsi atau dalam bentuk *ciphertext* akan memiliki ukuran yang sama dengan data asli atau *plaintext*, keuntungan lain dari menggunakan metode ini adalah proses enkripsi dan dekripsi yang sangat cepat sehingga tidak membuang banyak waktu.

Dari informasi tersebut, akan di rancang sebuah sistem penyimpanan data yang menggunakan metode algoritma *Rivest Code 4* (RC4) untuk mengamankan *database* perusahaan yang akan di implementasikan dalam Tugas Akhir dengan judul "Implementasi Keamanan Database Menggunakan Kriptografi RC4 Pada Database Sistem Iinformasi Berbasis Web Milik PT. TOROP SUMBER MAKMUR"

2. METODE PENELITIAN

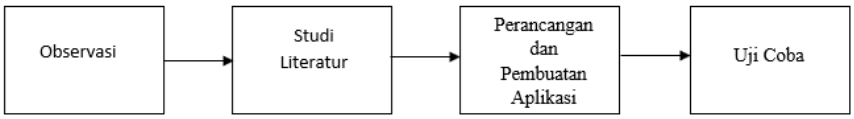
2.1 Data Penelitian

Data yang di gunakan pada penelitian ini adalah *Dataset* yang di peroleh dari beberapa proyek milik PT. Torop Sumber Makmur yang di adakan pada tahun 2017-2020. *Dataset* ini berisikan tentang penjualan, barang, nama *customer*, alamat *customer*, detail proyek, detail order, nilai kontrak, dan juga *service* yang di miliki oleh perusahaan itu sendiri.

2.2 Metode Pembanding

Pada penulisan di dalam sebuah penelitian, tentu banyak membutuhkan referensi dan juga sumber acuan yang memiliki tujuan tuntuk memberikan informasi kepada pembaca bahwa penulisan ini berasal dari sumber yang benar dan juga dapat di percaya. Dari beberapa jurnal yang ada, penulis menemukan perbandingan dari jurnal yang memiliki judul "Kriptografi Simetris RC4 Pada Transaksi Online Booking Engine System" (Ketut Agus Saputra dan Gede Arna Jude Saskara, 2020) yang mendapatkan rata-rata waktu enkripsi 5.5116 *milisecond*, sedangkan rata-rata waktu enkripsi yang di buat oleh penulis adalah 0.356941236 *milisecond*.

2.3 Penerapan Metode



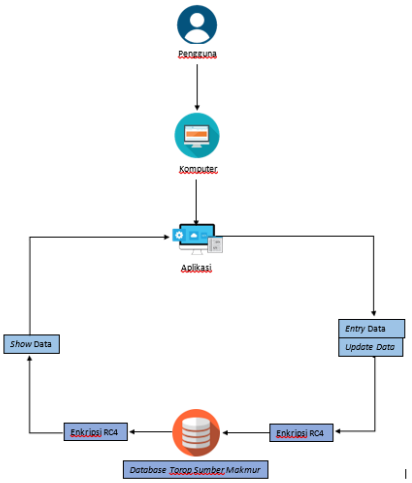
Gambar 1. Skema Penerapan Metode

Metode yang di gunakan adalah metode observasi yang merupakan metode yang di gunakan untuk mendapatkan semua data yang di perlukan pada penelitian ini yang nantinya data tersebut akan di gunakan untuk membangun system yang di perlukan untuk mengamankan data tersebut, dan langkah-langkahnya adalah sebagai berikut ini:

a. Observasi
   Pada tahap ini, penulis melakukan pengumpulan data dari perusahaan dan mengamati sistem apa yang di gunakan sebelumnya.
b. Studi Literatur
   Di tahap ini, penulis melakukan perbandingan dari penelitian yang sebelumnya telah di lakukan.
c. Perancanan dan Pembuatan Aplikasi
   Setelah semua data yang di kumpulkan serta di perlukan telah lengkap, maka akan di rancang dan di buat aplikasi kriptografi yang menggunakan RC4 sebagai algoritma enkripsinya.
d. Uji Coba
   Pada tahap ini, akan di lakukan sebuah pengujian kepada aplikasi yang telah di bangun dengan harapan akan berjalan sesuai dengan ekspektasi penulis dan juga mudah di gunakan bagi penggunanya. Pada tahap pengujian ini tidak terdapat kendala sama sekali dan sudah berjalan seperti apa yang di harapkan oleh penulis.

2.4 Arsitektur sistem

Arsitektur dari suatu sistem dapat memberikan gambaran secara garis besar tentang proses keseluruhan dari suatu ststem. Berikut ini adalah arsitektur system yang di gambarkan pada gambar 2 berikut ini.

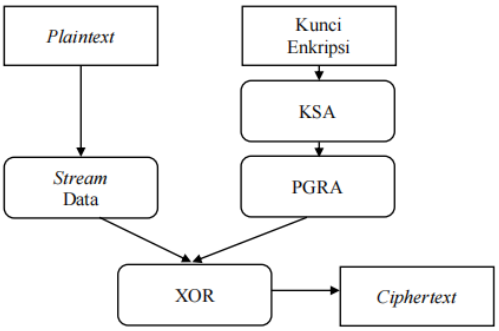


Gambar 2. Arsitektur Sistem

2.5 Algoritma RC4 *Stream Cipher*

RC4 menggunakan dua buah kotak substitusi (S-Box) array 256 byte yang berisi permutasi dari bilangan 0 sampai 255 dan S-Box kedua yang berisi permutasi fungsi dari kunci dengan panjang yang variabel. Cara kerja algoritma RC4 yaitu inisialisasi Sbox pertama, S[0],S[1],...,S[255], dengan bilangan 0 sampai 255.Pertama isi secara berurutan S[0] = 0, S[1] = 1,..., S[255] = 255. Kemudian inisialisasi array lain (S-Box lain), misal array K dengan panjang 256. Isi array K dengan kunci yang diulangi sampai seluruh array K[0], K[1],...,K[255] terisi seluruhnya. Dody Ronald saragi (2020). Dalam algoritma enkripsi, metode ini akan membangkitkan sebuah

pseudo random bytes dari *key* yang akan di gunakan dalam operasi XOR terhadap *Plaintext* untk dapat menghasilkan sebuah *ciphertext.* Berikut ini adalah diagram roses dari prosesi enkripsi sebuah data yang menggunakan algoritma RC4 *stream cipher* pada gambar 3.



Gambar 3. Rangkaian Proses Enkripsi RC4 *Stream Cipher*

Secara garis besar, RC4 stream cipher in terbagi menjadi dua tahapan, yaitu :

a. Tahap key scheduling (KSA), yaitu suatu state automaton yang di beri nilai awal berdasarkan kunci enkripsi.
b. Tahap pseudo-random generation (PRGA), yaitu suatu state automaton yang beroperasi dan outputnya menghasilkan keystream dan proses XOR dengan stream data.

Berikut ini, akan di jelaskan tentang Langkah-langkah dari algoritma RC4 stream cipher yang akan di jabarkan di bawah ini.

a. Key Scheduling Algorithm (KSA)

1. Inisialisasi S-Box (Array S). Proses dari inisialisasi sebuah S-Box (Array S) dapat di lihat pada tabel 1 di bawah ini :

```
for    i = 0 to 255
S[i] = i                                                    (1)
endfor
```

2. Inisialisasi S-Box (Array K). Proses dari inisialisasi sebuah S-Box (Array K) dapat di lihat pada Tabel 2 berikut ini :

```
for    i = 0 to 255
K[i] = kunci [i mod keylenght]                              (2)
endfor
```

3. Lalu, akan di lakukan suatu langkah pengacakan S-Box yang dapat di lihat pada Tabel 3 sebagai berikut :

```
i = 0; j = 0;
for i = 0 to 255;
j = (j + S[i] + K[i]) mod 256                               (3)
swap S[i] dan S[j]
endfor
```

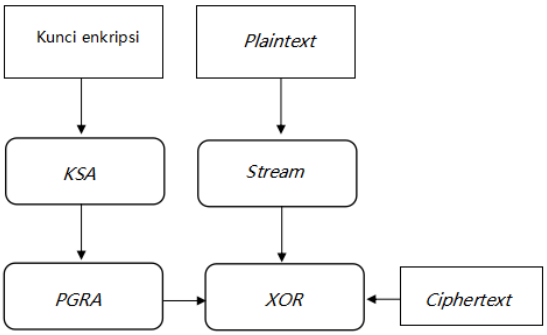
b. Pseudo Random Generation Algorithm (PRGA)

Setelah itu, buat pseudo random byte pada Tabel 4 seperti yang tertera di table berikut ini:

```
i = (i + 1) mod 256
j = (j + S[i]) mod 256
swap S[i] dan S[j]                                          (4)
t = (S[i] + S[j]) mod 256
K = S[t]
```

Maka, hasil akhir yang di dapatkan adalah sebuah ciphertext dengan hasil XOR mengantarkan stream key dari S-Box dan plaintext yang berurutan.

Algoritma dekripsi RC4 mirip dengan algoritma enkripsinya, perbedaannya hanya pada saat stream generation, yaitu untuk menghasilkan plainteks semula, maka ciphertext nya akan dikenakan operasi XOR terhadap pseudorandom bytenya. Algoritma key setup pada proses dekripsi sama dengan algoritma enkripsinya yang diproses inisialisasi S-Box, penyimpanan kunci kedalam key bytearray hingga proses inisialisasi S-Box berdasarkan key byte array nya. Untuk itu proses dekripsi dan enkripsi akan menghasilkan key stream yang sama. Perbedaannya hanya pada stream generationnya, yaitu yang dioperasikan bersama key stream adalah ciphertext untuk menghasilkan kembali plaintext. Harni Kusniyati, Satya Diansyah dan Raka Yusuf (2018). Proses Key Scheduling Algorithm (KSA) dilakukan sama halnya seperti proses enkripsi, dari mulai inisialisasi S-Box, hingga proses pengacakan S-Box. Untuk itu, proses dekripsi dan enkripsi akan menghasilkan key stream yang sama. Perbedaannya hanya pada stream generationnya, yaitu yang di operasikan bersama key stream adalah ciphertext untuk menghasilkan kembali plaintext yang sebelumnya di enkripsi. Berikut ini akan di berikan diagram proses dari suatu proses dekripsi data yang menggunakan algoritma RC4 Stream Cipher yang dapat dilihat pada gambar 4 berikut ini.



Gambar 4. Rangkaian Proses Dekripsi RC4 *Stream Cipher*

3. HASIL DAN PEMBAHASAN

Pada bagian ini akan di bahas tentang implementasi dari aplikasi yang di buat hingga waktu rata-rata dalam men-enkripsi dan juga dekripsi dari data.

3.1 Analisa Sistem

Adapun analisa kebutuhan sistem adalah sebagai berikut :

a. Sistem yang di buat dapat memberikan fitur otentikasi *user* melalui proses *login*.
b. Sistem yang di buat dapat memberikan fitur enkripsi (pengacakan data).
c. Sistem yang di buat dapat memberikan fitur dekripsi (mengembalikan data seperti semula).
d. Sistem yang di buat dapat memberikan fitur yang di butuhkan oleh perusahaan.

3.2 Pengujian

Data yang akan di gunakan dalam pengujian ini adalah data dari customer yang akan di jadikan contoh pada bagian ini, data tersebut dapat di lihat pada tabel 1 berikut ini.

Tabel 1. Tampilan data sebelum di enkripsi

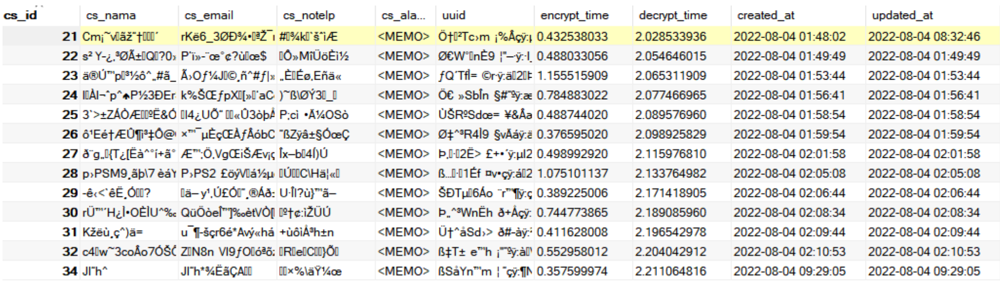
Sebelum Di Enkripsi			
NamaCustomer	Alamat	No Telp	Email
PT. PLN Persero	Jl. Peta Selatan No.39 6, RT.6/RW.3, Kalideres, West Jakarta City, Jakarta 11840	0215447630	ardhygunawan@staff.pln.co.id
PT. KRAKATAU DAYA LISTRIK	Krakatau Industrial Estate, Jl. Amerika	0254315001	staff@kdl.co.id

	No.I, Samangraya, Kec. Citangkil, Kota Cilegon, Banten 42443		
PT. Pertamina Persero	Building, Oil Centre, Lantai 1 - 4, Jl. M.H. Thamrin No.55, RT.9/RW.5, Gondangdia, Kec. Menteng, Kota Jakarta Pusat, Daerah Khusus Ibukota Jakarta 10350	02131906825	wahyu_andhika@staff.pertamina.com
PT.UNITED CAN COMPANY	Daan Mogot Rd No.KM. 17 9 2, RT.9/RW.2, Semanan, Kalideres, West Jakarta City, Jakarta 11850	0215447635	staff@unitedcan.com

Ketika data yang ada pada tabel di atas di *enkripsi* oleh sistem lalu kemudian di simpan kedalam *database*, maka data tersebut akan menjadi sebuah *ciphertext* seperti yang akan di jelaskan pada pembahasan berikut ini.

a. Enkripsi

Pada proses ini yang di ambil dari Tabel 1, ketika pengguna hendak memasukan data baru ke dalam *database*, maka data tersebut akan melalui proses enkripsi terlebih dahulu sebelum akhirnya di simpan ke dalam *database*. Sebagai contoh, berikut adalah data *customer* yang telah di enkripsi sebelum di masukan ke dalam tabel *customers* seperti yang tertera pada gambar 5 berikut.

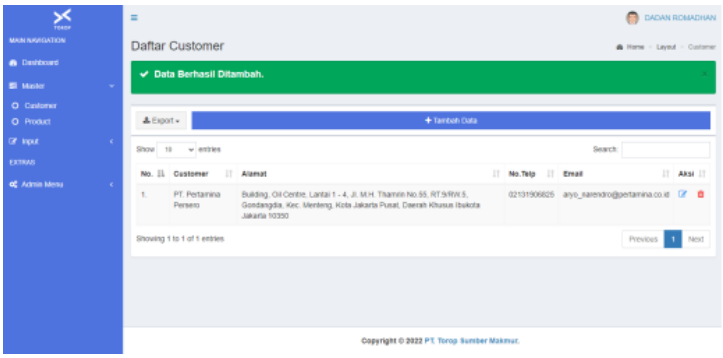


cs_id	cs_nama	cs_email	cs_notelp	cs_ala...	uuid	encrypt_time	decrypt_time	created_at	updated_at
21	Cm¡˜vlãž¨†'	rKë6_3ØÐ¾•ªŽ¯	#¾k`š˜iÆ	<MEMO>	Ö†²Tc›m ¡%Åçÿ:	0.432538033	2.028533936	2022-08-04 01:48:02	2022-08-04 08:32:46
22	s² Y-¿.³ØÃ±Q?0	P'ï»-¨œ°¢?ùœ$	Ô»MîÜöÈì½	<MEMO>	Ø€W°nÈ9 ¦"–ÿ:I	0.488033056	2.054646015	2022-08-04 01:49:49	2022-08-04 01:49:49
23	ä®Ú™p³½ô^„#ã_	Ã›Oƒ¼J©¸ñ^#ƒ|»	„ÈÉø,Eñä«	<MEMO>	ƒQ'Tfl= ©r·ÿ:ä2	1.155515909	2.065311909	2022-08-04 01:53:44	2022-08-04 01:53:44
24	ÅI¬ˆp^♠P½3ÐEr	k%ŠŒƒpX[»'aC	)˜ß\ØÝ3_	<MEMO>	Ö€ »SbÎn §#"ºÿ:æ	0.784883022	2.077466965	2022-08-04 01:56:41	2022-08-04 01:56:41
25	3`>±ZÁÒÆºË&Ó	l4¿UÕ" «Û3òþÅ	P;cì •Ä¼OSò	<MEMO>	ÙŠRºSdœ= ¥&Åa	0.488744020	2.089576960	2022-08-04 01:58:54	2022-08-04 01:58:54
26	ô¹Eé†ÆÛ¶ì³‡Ô@	×™¯µÈçŒÀƒÅóbC	¨ßZÿâ±§ÓœÇ	<MEMO>	Ø‡^³R4Ì9 §vÄáÿ:ä	0.376595020	2.098925829	2022-08-04 01:59:54	2022-08-04 01:59:54
27	ð¨g„{T¿[Ëà^°í+ã°	Æ™;Ö,VgŒiŠÆv¡ç	Îx–b4Í)Ú	<MEMO>	Þ,·2Ë> £+•´ÿ:µl2	0.498992920	2.115976810	2022-08-04 02:01:58	2022-08-04 02:01:58
28	p›PSM9¸ãþ\7 èáY	P›PS2 £öÿVá½µ	ÚC\Hä¦«	<MEMO>	ß…·1Éf ¤v•çÿ:á2	1.075101137	2.133764982	2022-08-04 02:05:08	2022-08-04 02:05:08
29	-ê‹<`êË¸Ó?	ä–y¹,Ú£Ó"¸®Áð:	U·Ì?ù}™ã–	<MEMO>	ŠÐTµ6Áo ¨r"¶ÿ:ç	0.389225006	2.171418905	2022-08-04 02:06:44	2022-08-04 02:06:44
30	rÜ™´H¿Ì•OÈÌU^‰	QüÖòeÎ™]‰ètVÒ	º†¢:ìŽÜÚ	<MEMO>	Þ„^³WnËh ð+Åçÿ:	0.744773865	2.189085960	2022-08-04 02:08:34	2022-08-04 02:08:34
31	Kžëù¸ç^)ä=	u¯¶-šçr6é*Avý«há	+ùôìÅ³h±n	<MEMO>	Ü†^åSd›> ð#-àÿ:	0.411628008	2.196542978	2022-08-04 02:09:44	2022-08-04 02:09:44
32	c4w˜3coÅo7ÓŠ	ZN8n VI9ƒOóªõ:	ReC}Õ	<MEMO>	ß‡T± e™h ¡""ºÿ:à	0.552958012	2.204042912	2022-08-04 02:10:53	2022-08-04 02:10:53
34	JI"h^	JI"h*¾ËãÇA	×%\äŸ¼œ	<MEMO>	ßSåYn™m ¦ ¨çÿ:¶N	0.357599974	2.211064816	2022-08-04 09:29:05	2022-08-04 09:29:05

Gambar 5. Tampilan data *customers* yang di enkripsi.

b. Dekripsi

Pada proses ini, ketika pengguna hendak melihat data yang baru di tambahkan atau yang sudah ada di dalam *database*, maka data tersebut akan melalui proses dekripsi terlebih dahulu sebelum akhirnya di tampilkan.pada halaman yang telah di tentukan.



Gambar 6. Tampilan data *customers* yang di dekripsi.

Pada sistem ini terdapat pencatatan untuk hasil dari enkripsi pada field "encrypt_time" dan juga hasil dari dekripsi pada *field* "decrypt_time". Pada masing-masing *field* system akan mencatat secara otomatis setiap pengguna memasukan data-data baru atau mengubah data. Khusus untuk "decrypt_time", *field* ini akan mencatat tiap salah satu halaman di akses secara otomatis.

a. Tabel *Customers*

Dari gambar 5, dapat di ketahui bahwa di dalam *database* ada *field* yang berisikan waktu dekripsi dan juga enkripsi dari aplikasi yang di buat, maka dapat di ketahui hasil dari "encrypt_time" dan juga "decrypt_time" dari masing-masing *record* yang ada yaitu:

Tabel 2. Tabel Waktu Rata-Rata
Enkripsi dan Dekripsi *record* Pada Tabel *customers*

No.	encrypt_time	decrypt_time
1.	0.275080919	0.526017904
2.	0.279503822	0.542235851
3.	0.550177097	0.552417994
4.	0.309747934	0.566122055
5.	0.248928070	0.575034857
6.	0.252695084	0.583338976
7.	0.271897078	0.591865063
8.	0.387934923	0.601594925
9.	0.700832129	0.609081030
10.	0.611579895	0.620672941
11.	0.495640039	0.631192923
12.	0.276774168	0.650717020
13.	0.272809982	0.658488989
Waktu Rata-rata	**0.364329326**	**0.630262005**

b. Tabel *Products*

Dari aplikasi yang di buat, maka di ketahui hasil dari "encrypt_time" dan "decrypt_time" dari masing-masing *record* yang ada yaitu:

Tabel 3. Tabel Waktu Rata-Rata
Enkripsi dan Dekripsi *record* Pada Tabel *products*

No.	encrypt_time	decrypt_time
1.	0.276015	0.513823
2.	0.406907	0.523766
3.	0.25	0.528735
4.	0.35	0.532627
5.	0.24	0.536692
6.	0.27	0.540753
7.	0.25	0.54604
8.	0.28	0.551111

9.	0.25	0.554663
10.	0.28	0.558693
11.	0.53	0.564145
12.	0.276015	0.513823
13.	0.406907	0.523766
Waktu Rata-rata	**0.307538**	**0.541004**

c. Tabel *Projects*

Dari aplikasi yang di buat, maka di ketahui hasil dari "encrypt_time" dan "decrypt_time" dari masing-masing *record* yang ada yaitu:

Tabel 4. Tabel Waktu Rata-Rata
Enkripsi dan Dekripsi *record* Pada Tabel *projects*

No.	encrypt_time	decrypt_time
1.	0.973625	0.572434
2.	0.278312	0.596136
3.	0.401265	0.60302
4.	0.304653	0.610065
5.	0.477844	0.616354
6.	0.290966	0.623167
Waktu Rata-rata	0.42674361	0.607494627

d. Tabel *Progress*

Dalam gambar 4.42 di atas, maka di ketahui hasil dari "encrypt_time" dan "decrypt_time" dari masing-masing *record* yang ada yaitu:

Tabel 5. Tabel Waktu Rata-Rata
Enkripsi dan Dekripsi *record* Pada Tabel *progress*

No.	encrypt_time (millisecond/byte)	decrypt_time (millisecond/byte)
1.	0.306567192	0.326750994
2.	0.242850065	0.340322018
3.	0.315689087	0.331357956
4.	0.492110968	0.33522892
Waktu Rata-rata	0.42674361	0.607494627

e. Tabel *Order_items*

Dari aplikasi yang di buat, maka di ketahui hasil dari "encrypt_time" dan "decrypt_time" dari masing-masing *record* yang ada yaitu:

Tabel 6. Tabel Waktu Rata-Rata
Enkripsi dan Dekripsi *record* Pada Tabel *order_items*

No.	encrypt_time	decrypt_time
1.	0.263972998	0.357729197
2.	0.265558958	0.271721125
3.	0.260847092	0.263219833
4.	0.247199059	0.260747194
5.	0.259180069	0.366134167
Waktu Rata-rata	0.259351635	0.303910303

f. Tabel rata-rata Waktu dari Semua Tabel

Berdasarkan penghitungan nilai rata-rata waktu enkripsi dan dekripsi dari masing-masing tabel yang ada di dalam *database*, maka di dapatkan hasil sebagai berikut ini:

Tabel 7. Tabel Waktu rata-rata Enkripsi dan Dekripsi *record* Pada *Database*

No.	Nama Tabel	Nilai rata-rata enkripsi (ms)	Nilai rata-rata dekripsi (ms)	Size (byte)
1.	*Customers*	0.364329326	0.630262005	8244
2.	*Products*	0.307538	0.541004	2897
3.	*Projects*	0.42674361	0.607494627	2086
4.	*Progress*	0.42674361	0.607494627	940
5.	*Order_items*	0.259351635	0.303910303	959
Waktu Rata-rata		**0.356941236**	**0.538033112**	**3025.2**

Berdasarkan data percobaan di atas yang di ambil dari Tabel 7, maka dapat di ambil sebuah kesimpulan bahwa aplikasi ini telah berhasil melakukan seluruh percobaan dan berjalan dengan semestinya, untuk waktu rata-rata yang di butuhkan dalam mengenkripsi sebuah record adalah 0.356941236 millisecond, 3025.2 byte, sedangkan waktu rata-rata yang di butuhkan untuk mendekripsi suatu record adalah 0.538033112 millisecond, 3025.2 byte.

4. KESIMPULAN

Jika di lihat dari uraian yang sebelumnya, maka didapat kesimpulan yaitu: Proses enkripsi dan juga dekripsi yang di lakukan pada database sistem yang di bangun telah berhasil di lakukan. Isi record dari masing-masing tabel dapat di enkripsi dan di dekripsi kembali. Jumlah atau panjang karakter dan juga nilai asli pada record yang terenkripsi tidak mengalami perubahan. Berdasarkan pengujian waktu enkripsi dan dekripsi sebelumnya, algoritma RC4 dapat melakukan proses enkripsi dan dekripsi dengan rata-rata waktu yang tergolong sangat cepat. Semakin panjang kunci untuk proses enkripsi maka semakin kuat keamanan enkripsi datanya. Dari hasil percobaan didapatkan rata-rata durasi proses enkripsi sebuah record adalah 0.15 detik dan rata-rata durasi proses dekripsi sebuah record adalah 0.27 detik.

DAFTAR PUSTAKA

[1] I. Afrianto and N. Taliasih, "*Sistem Keamanan Basis Data Klien PT Infokes Menggunakan Kriptografi Kombinasi RC4 Dan Base64*," *Jurnal Nasional Teknologi dan Sistem Informasi*, vol. 6, no. 1, pp.9-18, 2020.

[2] H. Kusniyati, S. Diansyah and R. Yusuf, "*Penerapan Algoritma Rivert Code 4 (Rc 4) Pada Aplikasi Kriptografi Dokumen*," *Jurnal PETIR*, vol. 11, no. 1, pp.38-47, 2018.

[3] N. Ratama and M. Munawaroh, "*Implementasi Metode Kriptografi dengan Menggunakan Algoritma RC4 dan Steganografi Least Significant Bit Dalam Mengamankan Data Berbasis Android*.," *Jurnal Media Informatika Budidarma*, vol. 6, no. 2, pp.1272-1281, 2022.

[4] A. Setiawan and T. Fatimah, "*Implementasi Algoritma Kriptografi RC4 Untuk Keamanan Database Aplikasi Penggajian Karyawan Berbasis Web Pada PT. Trans Intra Asia," SKANIKA*, vol. 4, no. 1, pp.66-71, 2021.

[5] F. S. Febriyani and A. Arfriandi, "Implementasi Algoritma RC4 pada Sistem Pengamanan Dokumen Digital Soal Ujian," *JISKA (Jurnal Informatika Sunan Kalijaga)*, vol. 6, no. 3, pp. 171–177, 2021.

[6] M. H. Hrp, N. B. Nugroho, and R. I. Ginting, "Implementasi Keamanan Data Gaji Pada Dinas Komunikasi Dan Persandian Kabupaten Aceh Tamiang Menggunakan Algoritma R*C4," Jurnal Cyber Tech*, vol. 1, no. 4, pp. 1-10, 2022.

[7] Z. Basim and P. Painem, "Implementasi Kriptografi Algoritma RC4 Dan 3DES dan Steganografi Dengan Algoritma EOF Untuk Keamanan Data Berbasis Desktop Pada SMK As-Su'udiyyah," *SKANIKA*, vol. 3, no. 4, pp.45-52, 2020.

[8] D. R. Saragi, et al, "Pengamanan Data File Teks (Word) Menggunakan Algoritma RC4," *Jurnal Sistem Komputer dan Informatika (JSON)*, vol. 1, no. 2, pp. 114-119, 2020.

[9] K. A. Seputra and G. A. J. Saskara, "*Kriptografi Simetris RC4 Pada Transaksi Online Booking Engine System*," *Jurnal Pendidikan Teknologi dan Kejuruan*, vol. 17, no. 2, pp.286-295, 2020.

[10] A. Andrico and M. Syafrullah, "*Aplikasi Enkripsi Database Menggunakan Algortima RC4 Berbasis Desktop*," *SKANIKA*, vol. 1, no. 3, pp.1011-1017, 2018.