3rd Seminar Nasional Mahasiswa Fakultas Teknologi Informasi (SENAFTI)
30 Agustus 2023 – Jakarta, Indonesia
Volume 2, Nomor 2, September 2023 - ISSN 2962-8628 (*online*)

APLIKASI KRIPTOGRAFI PENGAMANAN FILE MENGGUNAKAN ALGORITMA RC4 BERBASIS WEB PADA SMK MEDIA INFORMATIKA

Aditya Zulmar[1], Rizky Pradana[2]

[1,2]Teknik In*form*atika, Fakultas Teknologi Informasi, Universitas Budi Luhur, Jakarta, Indonesia

Email: [1]adityazulmar@gmail.com, [2*] rizky.pradana@budiluhur.ac.id
(* : corresponding author)

Abstrak- SMK Media In*form*atika adalah institusi pendidikan yang memiliki berbagai macam data penting, salah satunya adalah data soal ujian. Berkas yang berisi soal ujian adalah contoh pembobolan data yang terjadi di SMK Media In*form*atika. Untuk melindungi soal ujian perlu menggunakan teknologi keamanan data, maka dari itu penelitian ini bertujuan untuk mengamankan soal ujian dengan cara membuat aplikasi kriptografi pegamanan soal ujian agar data tidak bisa dilihat, diubah dan disebarkan oleh pihak yang tidak berwenang. Aplikasi kriptografi yang akan dibuat yaitu aplikasi PHP berbasis web pengamanan soal ujian dengan algoritma *Rivest Code 4* (RC4). Berdasarkan pengujian, aplikasi berjalan dengan baik dan soal ujian berhasil diamankan dengan metode enkripsi dengan aplikasi kriptografi yang dibuat dan dapat dikembalikan seperti semula dengan metode dekripsi tanpa mengalami perubahan pada *File size*. Hasil akhir pengujian rata-rata ukuran hasil proses *encrypt* sebesar 411648 *byte*, lamanya 131 *millisecond* dan rata-rata ukuran hasil proses *decrypt* sebesar 411648 *byte*, lamanya 131.1666666 *millisecond.*

Kata Kunci: Kriptografi, *Rivest Code 4* (RC4), Enkripsi, Dekripsi.

APPLICATION OF FILE SECURITY CRYPTOGRAPHY USING WEB-BASED RC4 ALGORITHM IN MEDIA INFORMATIKA VOCATIONAL SCHOOL

Abstract- *Media Informatika Vocational School is an educational institution that has various kinds of important data, one of which is exam question data. The file containing exam questions is an example of data breaches that occurred at SMK Media Informatika. To protect exam questions, it is necessary to use data security technology, therefore this study aims to secure exam questions by creating a cryptographic application to secure exam questions so that data cannot be seen, modified and disseminated by unauthorized parties. The cryptographic application that will be made is a web-based PHP application to secure exam questions with the Rivest Code 4 (RC4) algorithm. Based on the test, the application runs well and the exam questions are successfully secured by the encryption method with the cryptographic application made and can be returned to normal by the decryption method without changing the file size. The success rate of encryption and decryption is 100% with an average encryption time of 0.1314 and 0.1316 decryption.*

Keywords*: Cryptography, Rivest Code 4 (RC4), Encryption, Decryption*

1. PENDAHULUAN

Perkembangan teknologi komputer dan telekomunikasi di zaman sekarang telah mengubah cara masyarakat berkomunikasi. Salah satu kemajuan penting adalah pemanfaatan internet sebagai sarana pertukaran informasi. Namun, perlu diingat pentingnya menjaga keamanan informasi, mengingat internet digunakan oleh banyak orang. Aktivitas penyadapan dan manipulasi data bisa berdampak buruk bagi pengguna jaringan komunikasi saat ini. Karena itu, keamanan data menjadi hal yang sangat krusial[1].

Penggunaan dokumen digital saat ini semakin umum, termasuk dalam pembuatan soal ujian yang kini telah beralih menjadi *form*at data digital. Hal ini membawa manfaat kemudahan akses bagi guru dan siswa dalam menggunakannya[2]. Data merupakan salah satu asset terpenting yang perlu dilindungi karena tidak menutup kemungkinan data tersebut dapat bocor ataupun dicuri oleh orang yang akan menyalahgunakan data[3].

SMK Media In*form*atika merupakan salah satu institusi pendidikan yang pernah mengalami pembobolan data. Berkas yang berisi soal ujian adalah contoh pembobolan data yang terjadi di SMK Media

In*form*atika. Untuk melindungi soal ujian, perlu menggunakan teknologi keamanan data. Oleh karena itu, kriptografi diperlukan untuk mengatasi masalah tersebut melalui proses *encrypt* dan *decrypt*. *Encrypt* adalah proses penyandian *File* menggunakan algoritme khusus yang membuat *File* tidak dapat dibaca dengan mengubahnya menjadi kode yang kompleks. Proses pemulihan *File* terenkripsi dari kode kompleks ke kondisi semula dikenal sebagai dekripsi. Saat *File* dienkripsi, kunci diperlukan untuk mengamankannya, sehingga proses dekripsi kriptografi dapat mengembalikannya ke keadaan sebelumnya.

Maka dari itu, timbul inspirasi untuk mengatasi masalah yang terjadi yaitu membuat perangkat lunak berbasis web yang bisa menggunakan teknik kriptografi untuk melindungi arsip soal ujian. Algoritma RC4 akan diterapkan pada perangkat lunak kriptografi berbasis web yang akan dibuat. Perbedaan penelitian ini dengan penelitian yang sudah ada yaitu aplikasi ini benar-benar meng-enkripsi file, jadi ketika melakukan proses enkripsi tidak akan meninggalkan file yang belum dienkripsi (file hasil enkripsi akan disimpan dengan menimpa file asli).

Pertama kali didefinisikan sebagai ilmu menyandikan pesan. Tetapi saat ini kriptografi didefinisikan sebagai bidang yang menangani masalah keamanan data seperti kerahasiaan, integritas, dan otentikasi entitas dengan menggunakan teknik matematika. Oleh karena itu, kriptografi sekarang berarti lebih dari hanya menyembunyikan pesan, itu mencakup berbagai metode untuk melindungi sebuah informasi. Bersal dari bahasa Yunani, kriptografi diambil dari "*kryptos"* yaitu rahasia dan "*graphein"* yaitu menulis. Jadi, kriptografi memiliki arti menulis rahasia[4]. Dalam kriptografi ada dua proses, yaitu enkripsi dan deskripsi. Informasi untuk enkripsi disebut *plaintext*[5]. Pesan yang belum dibaca disebut ciphertext[6]. Dekripsi mengubah bentuk informasi tersandi menjadi informasi asli[7].

Algoritma RC4 menggunakan kunci simetris berupa *stream cipher*, yang melakukan proses unit dalam satu waktu. Unit atau data bisa berupa *byte* atau *bit*. 1987 RC4 ditemukan Ronald Rivest dan dijadikan simbol keamanan RSA[8]. RC4 merupakan algoritma aliran simetris eksklusif yang dikembangkan oleh RSA Data Security Inc (RSADSI)[9]. Berdasarkan tipe kuncinya, ada dau tipe kriptografi, yaitu simetris dan asimetris. RC4 salah satu contoh dari tipe simetris[10].

2. METODE PENELITIAN

2.1 Rivest Code 4 (RC4)

Berikut adalah penjelasan bagaimana perhitungan dan proses algoritma RC4 secara sederhana. Teks yang akan dienkripsi adalah ADIT dengan key 2537. Pertama lakukan proses KSA untuk menghasilkan *block Array* untuk penentuan kunci. Kemudian *Array* = S dan K dibentuk seperti:

S = 0, 1, 2, 3
K = 2, 3, 5, 7
i dan j adalah 0 lalu lakukan KSA untuk membentuk *Array* = secara acak:
Iterasi 1:
i = 0
j = (0+S[i]+K[i] modulo4
j = (0+0+2) modulo4 = 2
Swap S[0], S[2]
Array = S 2, 1, 0, 3

Iterasi 2:
i = 1
j = (2+S[i]+K[i] modulo4
j = (2+1+5) modulo4 = 0
Swap S[1], S[0]
Array = S 1,2, 0, 3

Iterasi 3:
i = 2
j = (0+S[i]+K[i] modulo 4
j = (0+0+7) modulo 4 = 3
Swap S[2], S[3]
Array = S 1,2, 3, 0

Iterasi 4:
i = 3
j = (3+S[i]+K[i] modulo4
j = (3+0+3) modulo4 = 2
Swap S[3], S[2]
Array = S 1, 2, 0, 3

Hasil akhir dari KSA digunakan pada langkah selanjutnya yaitu PRGA (Pseudo-Random Generation Algorithm) seperti:

Hasil KSA,
Array = S 1, 2, 0, 3
Iterasi 1:
i = 0
j = 0
i = (0+1) modulo4 = 1
j = (0+S[1]) modulo4 = 2
Swap S[1], S[2]
S = 1, 0, 2, 3
t = (S[1]+S[2]) modulo4
t = (0+2 modulo4)
t = 2
K1 = S[t] = S[2] = 2 = 00000010

Iterasi 2:
i = (1+1) modulo4 = 2
j = (2+S[2]) modulo4 = 0
Swap S[2], S[0]
S = 2, 0, 1, 3
t = (S[2]+S[0]) modulo 4
t = (1+2 modulo4)
t = 3
K2 = S[t] = S[3] = 3 = 00000011

Iterasi 3
i = (2+1) modulo4 = 3
j = (0+S[3] modulo4 = 3
Swap S[3], S[3]
S = 2, 0, 1, 3
t = (S[3]+S[3]) modulo4
t = (3+3 modulo4)
t = 2
K3 = S[t] = S[2] = 1 = 00000001

Iterasi 4
i = (3+1) modulo4 = 0
j = (3+S[0] modulo4 = 1
Swap S[0], S[1]
S = 0, 2, 1, 3
t = (S[0]+S[1]) modulo4
t = (0+2 modulo4)
t = 2
K4 = S[t] = S[2] = 1 = 00000001

Selanjutnya dilakukan proses XOR antara kode ASCII dari ADIT dengan nilai K1, K2, K3, K4. Berikut kode ASCII dari teks ADIT.

A = 01000001
D = 01000100
I = 01001001

T = 01010100

Jika kode ASCII untuk setiap karakter sudah didapat, maka dilakukan proses enkripsi XOR, sebagai berikut:

Plaintext : 01000001 (65) 01000100 (68) 01001001 (73) 01010100 (84)
Key : 00000010 (2) 00000011 (3) 00000001 (1) 00000001 (1)
Ciphertext : 01000011 (67) 01000111 (71) 01001000 (72) 01010101 (85)

Proses dekripsi juga menggunakan proses XOR seperti saat enkripsi, sebagai berikut:

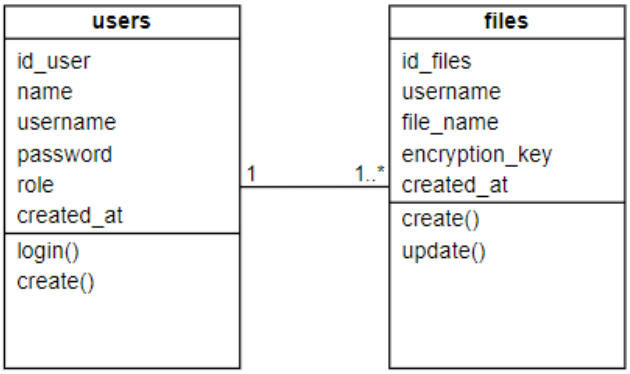
Key : 00000010 (2) 00000011 (3) 00000001 (1) 00000001 (1)
Ciphertext : 01000011 (67) 01000111 (71) 01001000 (72) 01010101 (85)
Plaintext : 01000001 (65) 01000100 (68) 01001001 (73) 01010100 (84)

Jadi setelah melakukan enkripsi maka teks ADIT menjadi kode biner 01000011 01000111 01001000 01010101, pada kode ASCII menjadi CGHU.

2.2 Rancangan Basis Data

a. Class Diagram

Berperan dalam memberikan gambaran mengenai struktur dan relasi antar objek yang tersedia. Struktur ini mencakup atribut dan metode yang terdapat dalam setiap class.. Class Diagram bisa pada gambar 1.



Gambar 1. Class Diagram

b. Spesifikasi Basis Data

Tabel *User*s
Nama Database : db_rc4
Nama Tabel : *User*s
PK : id_*User*

Table 1. Tabel *User*

No	Field	Tipe Data	Panjang	Keterangan
1	id_user	tinyint	4	Id Pengguna
2	name	varchar	100	Nama Pengguna
3	username	varchar	100	Username
4	password	varchar	100	Password/Kata Sandi
5	role	varchar	100	Level Pengguna
6	created_at	timestamp	-	Tanggal Dibuat

Tabel *File*s
Nama Database : db_rc4
Nama Tabel : *File*s
PK : id_*File*s

Table 2. Tabel *Files*

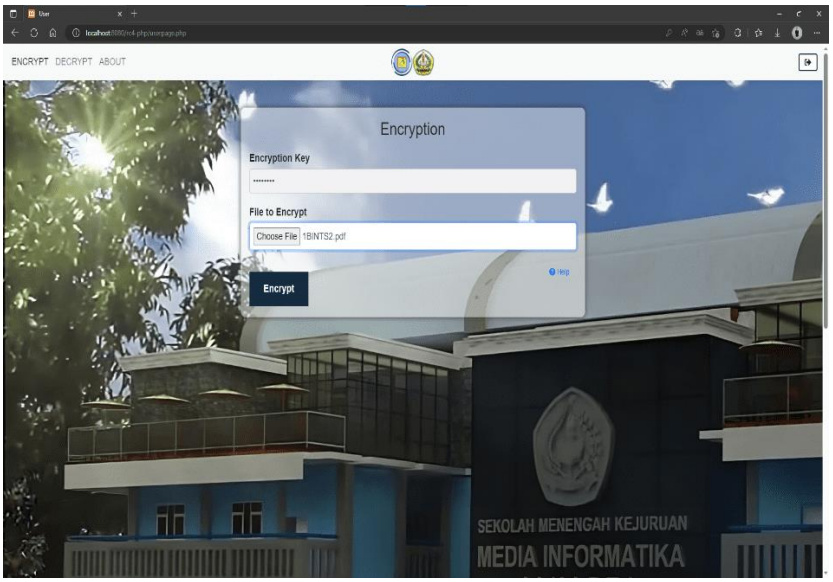
No	Field	Tipe Data	Panjang	Keterangan
1	id_*files*	tinyint	4	Id *File*
2	*username*	varchar	100	Nama Pengguna
3	*File*_name	varchar	100	Nama *File*
4	*encrypt*ion_key	varchar	100	Key Enkripsi
5	created_at	timestamp	-	Tanggal Dibuat

3. HASIL DAN PEMBAHASAN

3.1 Pengujian

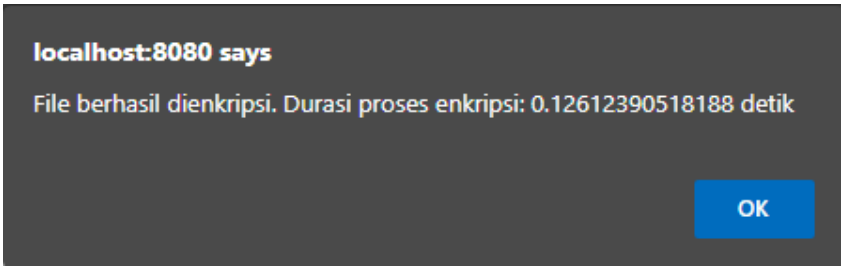
a. *Encrypt*

User harus mengisi *form* pada menu *encrypt* jika ingin melakukan enkripsi. Gambar 3 adalah tampilan layar *form encrypt* yang sudah diisi.



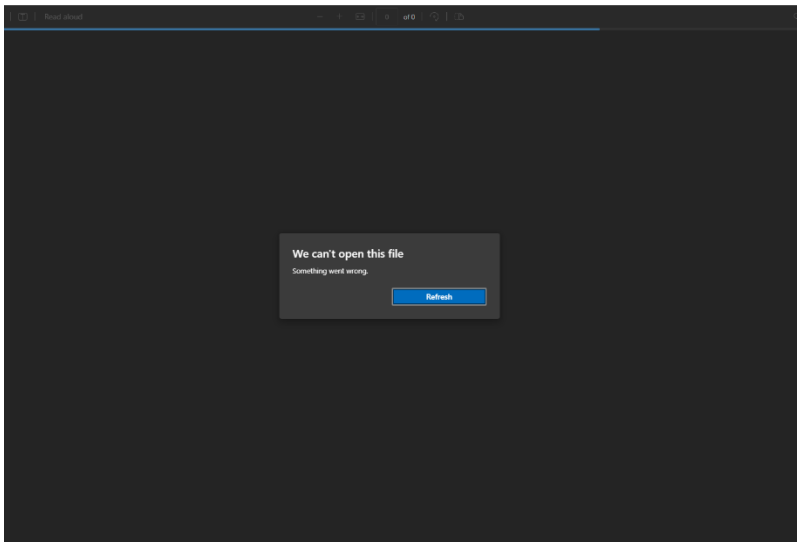
Gambar 2. Tampilan Layar *Encrypt*

Setelah mengisi *form*, klik *button Encrypt* untuk melakukan proses enkripsi. Jika berhasil akan tampil pesan seperti gambar 4.



Gambar 3. Pesan Berhasil Enkripsi

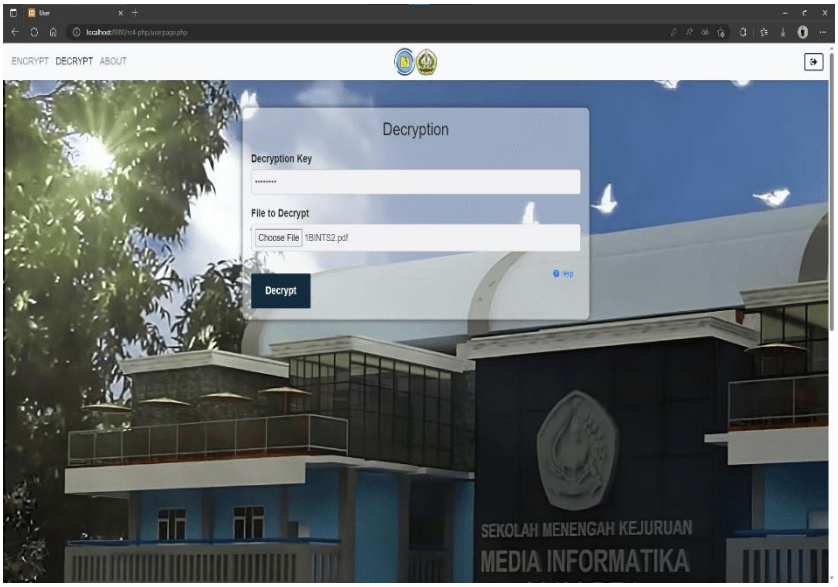
File yang sudah dienkripsi tidak akan bisa dilihat karena sudah terjadi perubahan. Gambar 5 adalah tampilan *File* setelah dienkripsi.



Gambar 4. *File* Sesudah Proses *Encrypt*

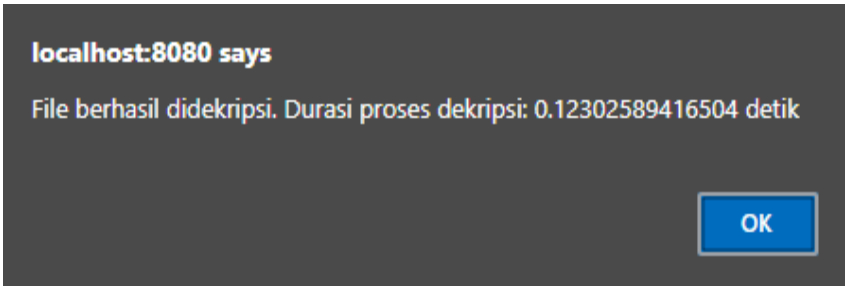
b. Decrypt

User harus mengisi *form* pada menu Decrypt jika ingin melakukan dekripsi. Gambar 6 adalah tampilan layar *form* decrypt yang sudah diisi.



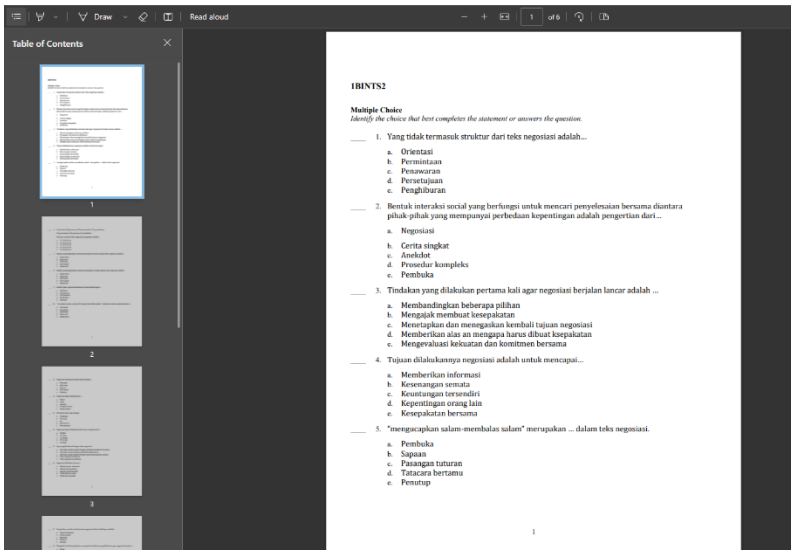
Gambar 5. Tampilan Layar *Decrypt*

Setelah mengisi *form*, klik *button* Decrypt untuk melakukan proses dekripsi. Jika berhasil akan tampil pesan seperti gambar 7.



Gambar 6. Pesan Berhasil Dekripsi

File yang sudah didekripsi akan dikembalikan ke keadaan semula, sehingga bisa dilihat kembali. Gambar 8 adalah tampilan *File* setelah didekripsi.



Gambar 7. *File* Sesudah Proses *Decrypt*

c. Tabel Pengujian

Pada tabel pengujian ini akan membandingkan ukuran *File* sebelum dan sesudah melakukan proses *encrypt* dan decrypt, durasi proses *encrypt* dan *decrypt*. Berikut tabel pengujian proses *encrypt* dan *decrypt* bisa dilihat pada tabel 3 dan 4.

Table 3. Pengujian Proses *Encrypt*

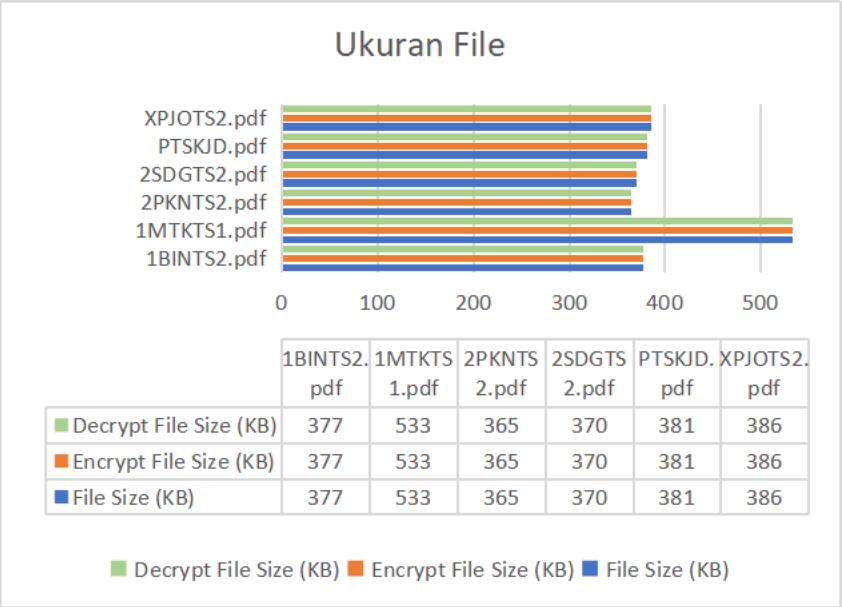
No	**Ukuran *File* Asli *(Byte)***	**Nama *File***	**Ukuran *File Encrypt (Byte)***	**Durasi *Encrypt (Millisecond)***
1	386048	1BINTS2.pdf	386048	126
2	545792	1MTKTS1.pdf	545792	170
3	373760	2PKNTS2.pdf	373760	116
4	378880	2SDGTS2.pdf	378880	119
5	390144	PTSKJD.pdf	390144	125
6	395264	XPJOTS2.pdf	395264	130
Rata-rata	411648		411648	131

Table 4. Table Proses *Decrypt*

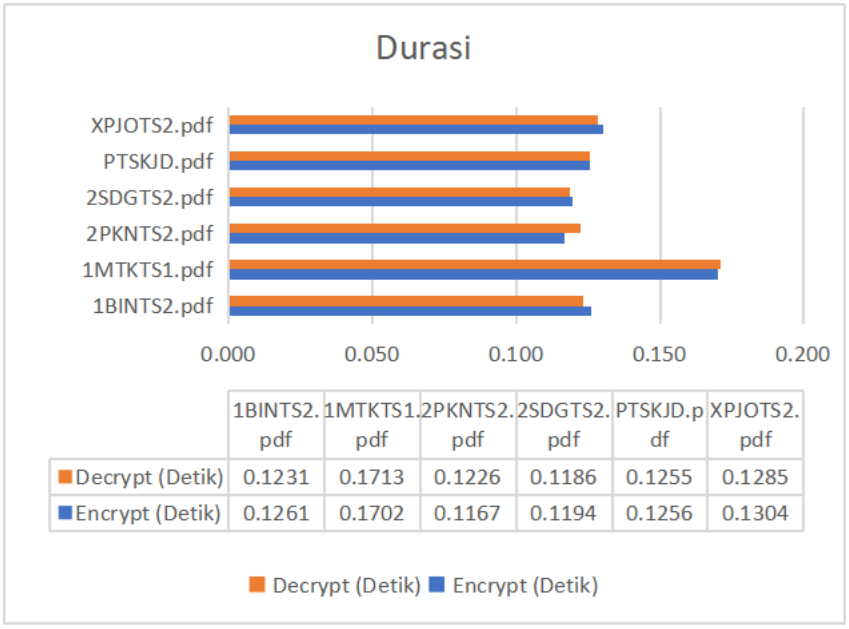
No	**Ukuran *File* Asli *(Byte)***	**Nama *File***	**Ukuran *File Decrypt (Byte)***	**Durasi *Decrypt (Millisecond)***
1	386048	1BINTS2.pdf	386048	123
2	545792	1MTKTS1.pdf	545792	171
3	373760	2PKNTS2.pdf	373760	122
4	378880	2SDGTS2.pdf	378880	118
5	390144	PTSKJD.pdf	390144	125
6	395264	XPJOTS2.pdf	395264	128
Rata-rata	411648		411648	131.1666666

Hasil akhir pengujian rata-rata ukuran hasil proses *encrypt* sebesar 411648 *byte*, lamanya 131 *millisecond* dan rata-rata ukuran hasil proses *decrypt* sebesar 411648 *byte*, lamanya 131.1666666 *millisecond*.

d. Grafik Pengujian
Pada tahap ini akan dibuat grafik pengujian berdasarkan data pada tabel pengujian untuk mempermudah melihat perbandingan pada setiap data. Berikut adalah grafik pengujian ukuran *File* dan grafik pengujian durasi bisa dilihat pada gambar 9 dan 10.



Gambar 8. Grafik Pengujian Ukuran *File*



Gambar 9. Grafik Pengujian Durasi

4. KESIMPULAN

a. Aplikasi berjalan dengan baik dan sesuai untuk memberikan perlindungan terhadap data soal ujian menggunakan algoritma RC4.
b. Data soal ujian dapat dienkripsi dengan aplikasi kriptografi yang telah dibuat.
c. Enkripsi dan dekripsi dengan algoritma RC4 tidak mengalami perubahan terutama dari segi ukuran *File*.
d. Tingkat keberhasilan enkripsi dan dekripsi 100% dengan rata-rata waktu enkripsi 131 *millisecond* dan dekripsi 131.1666666 *millisecond*.

DAFTAR PUSTAKA

[1] Aswita Raja Sari Novica, Gunawan Indra, Nasution Zulaini Masuro, Sumarno, and Tambunan Heru Satria, "IMPLEMENTASI ALGORITMA AES DAN RC4 TERHADAP KEAMANAN DATA PRODUK BENIH SAYURAN DI PT. EWINDO," *Jurnal Sosial Sains*, vol. 1, no. 6, pp. 461–468, 2021.

[2] F. S. Febriyani and A. Arfriandi, "Implementasi Algoritma RC4 pada Sistem Pengamanan Dokumen Digital Soal Ujian," *JISKa*, vol. 6, no. 3, pp. 171–177, 2021.

[3] Simatupang Leo Dearman and Khairil, "Pengamanan Dokumen Teks Dengan Menerapkan Kombinasi Algoritma Kriptografi Klasik," *Jurnal Teknik Informatika Unika St. Thomas (JTIUST)*, vol. 7, no. 1, pp. 133–140, 2022.

[4] R. A. Umar and S. Hari, "IMPLEMENTASI ALGORITMA RC4 UNTUK KEAMANAN *FILE* BERBASIS WEB PADA SDIT AR RAHMAN," *Seminar Nasional Mahasiswa Fakultas Teknologi Informasi (SENAFTI)*, vol. 1, no. 1, pp. 377–385, 2022, Accessed: Jul. 23, 2023. [Online]. Available: https://senafti.budiluhur.ac.id/index.php/senafti/index

[5] A. S. Y. Irawan, A. P. Pratama, and R. Antono, "RC4 Cryptography Implementation Analysis on Text Data," *ISSN*, vol. 11, no. 2, pp. 115–120, 2021, [Online]. Available: http://journal.stmikglobal.ac.id/index.php/sisfotek

[6] Wahyudi, D. Hartama, I. O. Kirana, Sumarno, and I. Gunawan, "Implementasi Algoritma Kriptografi Rivest Shamir Adlemen untuk Mengamankan Data Ijazah pada SMK Swasta Prama Artha Kab. Simalungun," *Jurnal Ilmu Komputer dan Informatika*, vol. 2, no. 1, pp. 57–66, Jan. 2022, doi: 10.54082/jiki.19.

[7] R. Suhardianto and J. Manurung, "Cryptography Application to Message Text using the Android- Based RC4 Method," *Jurnal Teknologi Komputer*, vol. 14, no. 2, pp. 190–197, 2020, [Online]. Available: http://login.seaninstitute.org/index.php/Loginˆ190Loginˆ Loginˆ190Journalhomepage:http://login.seaninstitute.org/index.php/Login

[8] Soleman, D. Budiman, and S. Mubaroq, "Combination RC4 Algorithm and Base64 *Encrypt*ion on The Least Significant Bit Method," *Jurnal CoreIT: Jurnal Hasil Penelitian Ilmu Komputer dan Teknologi Informasi*, vol. 8, no. 2, pp. 103–114, Jan. 2022, doi: 10.24014/coreit.v8i2.20106.

[9] R. Sulaiman, C. Kirana, T. Sugihartono, Laurentinus, and F. Panca Juniawan, "RC4 Algorithm and Steganography to Double Secure Messages in Digital Image," in *2020 8th International Conference on Cyber and IT Service Management, CITSM 2020*, Institute of Electrical and Electronics Engineers Inc., Oct. 2020. doi: 10.1109/CITSM50537.2020.9268833.

[10] Pramusinto Wahyu, Wizaksono Nugroho, and Saputro Ari, "Aplikasi Pengamanan *File* Dengan Metode Kriptografi AES 192, RC4 Dan Metode Kompresi Huffman," *JURNAL BIT (Budi Luhur Information Technology)*, vol. 16, no. 2, pp. 47–53, 2019, Accessed: Jul. 23, 2023. [Online]. Available: https://journal.budiluhur.ac.id/index.php/bit