4th Seminar Nasional Mahasiswa Fakultas Teknologi Informasi (SENAFTI)
27 Maret 2024 – Jakarta, Indonesia
Volume 3, Nomor 1, April 2024 - ISSN 2962-8628 (*online*)

PENERAPAN ALGORITMA RC6 UNTUK PENGAMANAN FILE LAPORAN KEUANGAN DI THE BELLAGIO MANSION BERBASIS WEB

Raditya Ananda Putra[1*], Rizky Pradana[2]

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

Email: [1*]radityap2107@gmail.com, [2]rizky.pradana@budiluhur.ac.id
(*: corresponding author)

Abstrak- The Bellagio Mansion beroperasi di sektor *real estate* yang menyediakan apartemen *strata title* dan lift pribadi. The Bellagio Mansion memiliki laporan data keuangan yang meliputi pendapatan, laba rugi, jumlah asset, dll. Data tersebut tersimpan dalam komputer perusahaan dan dikhawatirkannya data tersebut tercuri dan diakses oleh pihak yang tidak berwenang. Perusahaan tersebut harus memiliki aplikasi untuk mengamankan *file* agar tidak terjadi kebocoran data yang berakibat merugikan perusahaan tersebut. Dalam hal ini, The Bellagio Mansion memerlukan aplikasi untuk melindungi data perusahaannya. Aplikasi tersebut berupa enkripsi dan dekripsi dengan memakai metode RC6. Metode ini menggunakan blok 128 bit dibagi menjadi 4 buah register 32 bit dan menggunakan prinsip *Iterated Block Cipher* yang menggunakan iterasi dimana iterasi tersebut dilakukan sebanyak 20 kali putaran. Dengan adanya iterasi tersebut keamanan akan semakin terjamin, namun performa pada metode ini akan berkurang karena membutuhkan waktu yang lebih lama untuk melakukan iterasi. Tahapan pengujian pada aplikasi yaitu dengan memasukkan kunci untuk mengenkripsi sebuah *file* dan menggunakan kunci yang sama untuk mendekripsi *file*. Hasil pengujian yaitu enkripsi dan dekripsi berhasil dilakukan, juga mengubah ukuran *file* dan nama *file* serta *file* tidak bisa dibuka tanpa proses dekripsi. Pengujian menunjukkan bahwa metode RC6 dapat digunakan untuk menyimpan file penting perusahaan dengan aman.

Kata Kunci: Kriptografi, RC6, Enkripsi, Dekripsi, *File*

APPLICATION OF RC6 ALGORITHM TO SECURE FINANCIAL REPORT FILES AT THE BELLAGIO MANSION WEB-BASED

Abstract- *The Bellagio Mansion operates in the real estate sector which provides strata title apartments and private lift. The Bellagio Mansion has financial data reports which include revenue, profit and loss, total assets, etc. The data is put away within the company's computer and it is feared that the data is stolen and accessed by unauthorized parties. The company must have an application to secure files to prevent data leakage that results in harm to the company. In this case, The Bellagio Mansion needs an application to protect its company data. This application has the form of encoding and decoding using RC6 method. This method uses 128 bit block divided into 4 32 bit registers and uses the Iterated Block Cipher principle which uses iteration where the iteration is done 20 times round. With the iteration, security will be guaranteed, but the performance of this method will decrease because it takes longer to iterate. The testing phase of this application is by entering the key to encrypt a file and using the same key to decrypt the file. The test results are encryption and decryption successfully carried out, also changing the file size and file name and the file cannot be opened without the decryption process. Testing showed that the RC6 method can be used to safely store important corporate files.*

Keywords*: Cryptography, RC6, Encryption, Decryption, File*

1. PENDAHULUAN

Seiring kemajuan teknologi, ancaman terhadap keamanan data semakin beragam. Data yang disimpan dan diproses dalam berbagai organisasi maupun perusahaan memiliki nilai tertentu. Informasi tersebut dapat mencakup data pribadi pelanggan, rahasia bisnis, data medis, informasi keuangan dan masih banyak lagi. Nilai informasi tersebut menjadikannya target yang berharga bagi penjahat, pesaing, atau pihak jahat lainnya[1]. Untuk melindungi data dan informasi yang sensitif dan rahasia dari akses yang tidak sah yaitu dengan menggunakan teknik kriptografi[2]. *Crypto* dan *graphia* berasal dari Bahasa Yunani, masing-masing bermakna menyembunyikan dan menulis. Kriptografi merupakan bidang dalam hal keamanan yang menggunakan teknik matematika untuk mempertimbangkan hal seperti validitas, integritas, kerahasiaan, dan otentikasi data dalam hal keamanan informasi[3].

Salah satu metode yang tersedia dalam keamanan data adalah kriptografi. Kriptografi terbagi menjadi dua yaitu: simetris dan konvesional. Kriptografi simetris yaitu jenis algoritme yang membutuhkan kunci serupa pada prosedur enkripsi dan dekripsi. Suatu protokol pergeseran kunci di jalur publik diperlukan agar pergeseran kunci simetris tetap aman [4, 5].

The Bellagio Mansion didirikan pada tahun 2011 yang bergerak di bidang real estate yang menyediakan apartemen *strata title* dan lift pribadi. The Bellagio Mansion memiliki laporan data keuangan yang meliputi pendapatan, laba rugi, jumlah asset, dll. Data tersebut tersimpan dalam komputer perusahaan dan dikhawatirkannya data tersebut tercuri dan diakses oleh pihak yang tidak berwenang. Perusahaan tersebut harus memiliki aplikasi untuk mengamankan *file* agar tidak terjadi kebocoran data yang berakibat merugikan perusahaan tersebut.

Penelitian yang terkait dengan RC6 yaitu [6] membahas tentang kriptografi dengan menggunakan metode RC6 yang berjudul "Implementasi Algoritma RC6 Dalam Pengamanan File Pada BP2RD SU SAMSAT SEI RAMPAH" pada penelitian tersebut bertujuan untuk mengamankan file data kendaraan bermotor. Perbedaan dengan penelitian sebelumnya yaitu pada objek yang diuji dan juga penelitian ini menggunakan sistem berbasis web. Kontribusi penelitian yaitu dengan menambahkan beberapa ekstensi file yang bisa dienkripsikan seperti pdf, pptx, txt dan juga file gambar seperti jpg dan png.

Rivest Code 6 (RC6) merupakan algoritma kriptografi dengan efek *avalanche* plaintext dan kunci yang baik dan waktu yang cepat dalam proses enkripsi dan dekripsi[7]. Algoritme RC6 dipilih karena algoritme tersebut menghasilkan keamanan yang memadai tanpa mengurangi kemampuan aplikasi [8]. Oleh karena itu, algoritme RC6 merupakan pilihan yang tepat untuk mengamankan data laporan keuangan pada The Bellagio Mansion.

2. METODE PENELITIAN

2.1 Data Penelitian

Data yang digunakan dalam penelitian ini yaitu berupa *file* mengenai laporan keuangan yang ada di The Bellagio Mansion dalam ekstensi pdf, docx, dan xlsx.

2.2 Algoritme RC6

Algoritme RC6 merupakan algoritme dengan berbagai parameter utama: w, r, dan b. Dengan demikian, algoritme RC6 juga dapat dituliskan seperti berikut : RC6-w/r/b, yaitu w=32 mewakili besar ukuran pada bit dan r=20 mewakili bilangan bulat positif dan mewakili jumlah iterasi. Saat prosedur pengenkripsian berlangsung, b diatur antara 16, 24 atau 32 *byte* mewakili besar ukuran kunci pada *byte* [9].

Dengan mengikuti enam aturan pengoperasian dasar, RC6 menguraikan 128 bit blok menjadi empat 32 bit blok, diantaranya:

a. A + B : Pengoperasian jumlah bilangan bulat
b. A – B : Pengoperasian kurang bilangan bulat.
c. A ⊕ B : Pengoperasian XOR/Ekslusif-OR.
d. A x B : Pengoperasian kali bilangan bulat.
e. A<<<B : Nilai A diputar sebesar komponen B ke kiri.
f. A>>>B : Nilai A diputar sebesar komponen B ke kanan.

2.3 Penerapan Metode Algoritme RC6

Implementasi adalah penerapan proses eksekusi suatu sistem yang dibuat, khususnya sistem logika yang diimplementasikan dalam suatu sistem komputer (program) yang terstruktur, sehingga dapat diberikan kepada pengguna gambaran bagaimana menjalankan program tersebut hingga menghasilkan data yang diinginkan[6]. Penerapan metode RC6 meliputi 3 langkah yaitu:

2.3.1. Enkripsi

Dengan membagi 128-bit blok tersebut menjadi empat buah 32-bit blok, algoritme beroperasi pada empat 32-bit blok yang disebut A, B, C, dan D. Di blok A akan dipasang *byte* awal, dan di blok D akan dipasang *byte* akhir, sehingga (A, B, C, D) = (B, C, D, A). Dengan kata lain, blok di sebelah kanan bersumber dari blok di sebelah kiri [10]. Algoritme enkripsi RC6 dapat dilihat pada persamaan 1.

```
B = B + S [ 0 ]
D = D + S [ 1 ]
for i = 1 to 20 do {
t = ( B x ( 2B + 1 ) ) <<< 5
u = ( D x ( 2D + 1 ) ) <<< 5
A = ( ( A ⊕ t ) <<< u ) + S[ 2i ]
C = ( ( C ⊕ u ) <<< t ) + S[ 2i + 1 ]
(A, B, C, D) = (B, C, D, A)
}
A = A + S[ 42 ]
C = C + S[ 43 ]
```

(1)

Algoritme RC6 dimulai dari subkunci S[0] sampai S[43]. Panjang setiap subkunci adalah 32 bit. Dalam algoritme RC6, proses whitening dilakukan sebelum dan sesudah enkripsi. Ini dilakukan untuk mengaburkan iterasi awal dan akhir pada proses enkripsi dan dekripsi. Nilai B dimasukkan ke S[0] sedangkan nilai D dimasukkan ke S[1]. Disetiap iterasi, RC6 memakai dua subkunci: iterasi awal memakai subkunci S[2] dan S[3], dan iterasi kedua memakai subkunci selanjutnya. Pada saat iterasi akhir selesai, prosedur *whitening* akhir dilakukan. Nilai A di jumlahkan pada S[42] kemudian nilai C di jumlahkan pada S[43] [10].

Aturan berikut diterapkan pada setiap iterasi algoritme RC6: Nilai B disisipkan pada fungsi f(x) =x(2x+1), dan sebesar lg-w atau 5 bit akan diputar ke kiri, kemudian ditampilkan sebagai u. Pada nilai t juga dipakai pada nilai C dan memutarnya ke kiri, kemudian nilai u juga dipakai di nilai A dan memutarnya ke kiri. Selanjutnya, iterasi menambahkan subkunci S[2i] ke A, lalu subkunci S[2i+1] ke C. Kemudian, 4 blok tersebut ditukar sesuai aturan, sehingga blok A diletakkan di blok D, kemudian blok B diletakkan di blok A, lalu blok C diletakkan di blok B, dan blok D diletakkan di blok C, iterasinya bertahan sebanyak dua puluh kali [10].

2.3.2. Dekripsi

Algoritme RC6 mendekripsi ciphertext berbeda pada saat melakukan enkripsi. Pada saat enkripsi yang beroperasi adalah penjumlahan, Sebaliknya, pada saat dekripsi yang beroperasi adalah pengurangan. Subkunci yang dipakai untuk *whitening* pada tahap pertama akan digunakan untuk langkah kedua, dan sebaliknya, subkunci yang digunakan untuk *whitening* setelah langkah pertama akan diterapkan untuk langkah kedua. Oleh karena itu, untuk melakukan dekripsi, hanya perlu menggunakan algoritme yang sama enkripsi, menggunakan subkunci yang sama untuk enkripsi setiap kali, dan hanya membalik urutan subkunci yang digunakan [10]. Algoritme dekripsi untuk RC6 dapat dilihat pada persamaan 2.

```
C = C - S[ 43 ]
A = A - S[ 42 ]
for i = 20 downto 1 do
{
(A, B, C, D) = (D, A, B, C)
u = ( D x ( 2D + 1 ) ) <<< 5
t = ( B x ( 2B + 1 ) ) <<< 5
C = ( ( C - S[ 2i + 1 ] ) >>> t ) ⊕ u
A = ( ( A - S[ 2i ] ) >>> u) ⊕ t
}
D = D - S[ 1 ]
B = B - S[ 0 ]
```

(2)

2.3.3. Pembangkitan Kunci

Untuk melakukan pembangkitan yaitu dengan menginput kunci b *byte* $(0 \leq b \leq 255)$. Kunci pada *byte* ini disusun pada array c w-bit L[0]. L[c-1]. Kunci pada *byte* awal diletakkan di L[0] dan *byte* selanjutnya diletakkan di L[1], begitupun pada *byte-byte* berikutnya. (Perhatikan bahwa jika b=0, maka c=1 dan L[0]=0). Setiap kata W-bit dihasilkan dengan menambahkan round kunci 2r+4 dan diletakkan dalam array S[0,…,2r + 3] [10].

Konstanta P32=B7E15163 diambil dari ekstensi biner e 2 di mana e merupakan fungsi logaritma semetara itu konstanta Q32=9E3779B9 didapat melalui ekspansi biner ϕ -1. Di sini ϕ disebut dengan "Rasio Emas" Konstanta P32 dan Q32 merupakan "konstanta ajaib" pada pembangkitan kunci. [10]. Algoritme untuk menghasilkan kunci dapat dilihat pada persamaan 3.

```
S[ 0 ] = 0xB7E15163
for i = 1 to 43 do S[i] = S[i-1] + 0x9E3779B9
A = B = i = j = 0
for k = 1 to 132 do
{
A = S[ i ] = ( S[ i ] + A + B) <<< 3
B = L[ j ] = ( L[ j ] + A + B) <<< ( A + B )
i = ( i + 1 ) mod 44
j = ( j + 1 ) mod c
}
```
(3)

2.4 Rancangan Pengujian

Rancangan Pengujian yang akan dilakukan berupa *file* ekstensi, besar ukuran pada *file*, waktu untuk melakukan enkripsi, waktu untuk melakukan dekripsi, ukuran *file* setelah pengujian, dan hasil nama *file* yang telah dienkripsi maupun didekripsi.

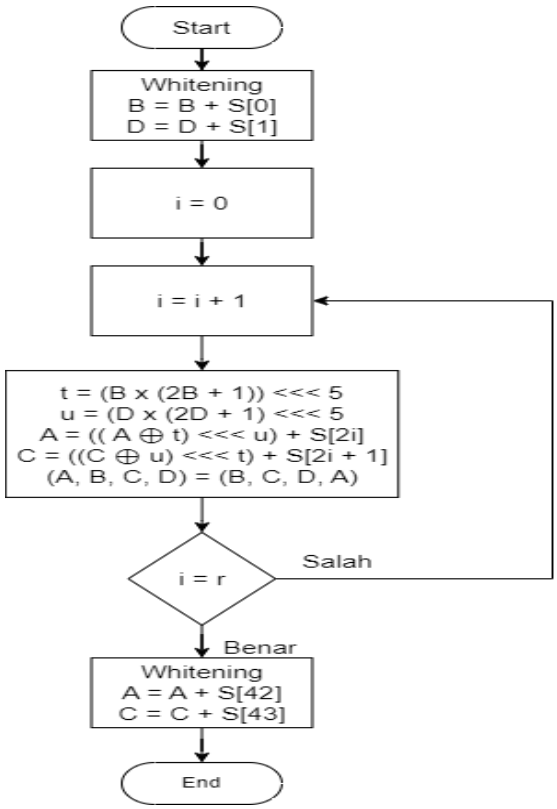
3. HASIL DAN PEMBAHASAN

3.1 *Flowchart*

Untuk menggambarkan urutan proses pada program ini akan menggunakan sebeagai representasi visual dari skema proses program.

3.1.1 *Flowchart* Enkripsi RC6

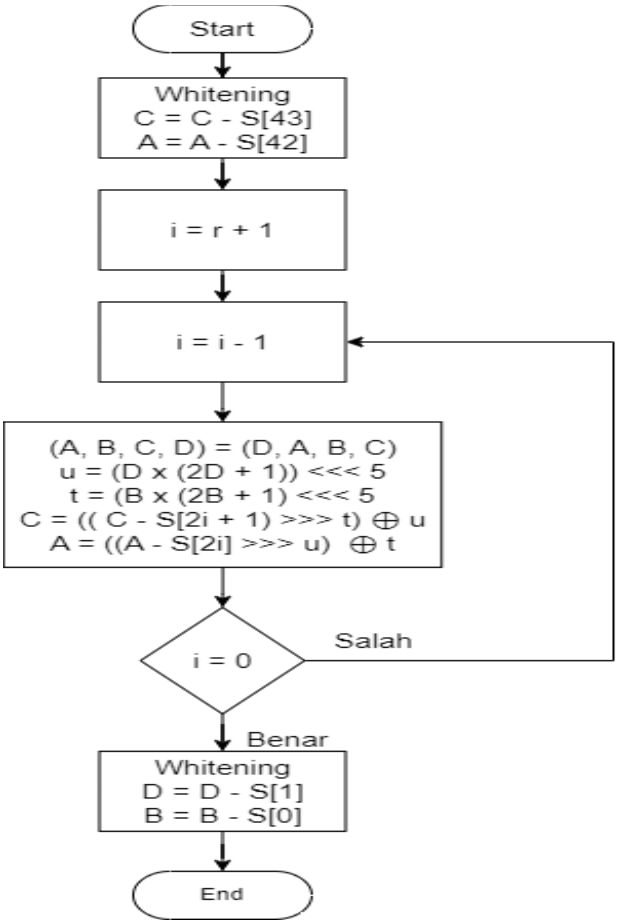
Pada gambar 1, merupakan *flowchart* yang digunakan untuk enkripsi RC6 dalam penelitian.



Gambar 1. *Flowchart* Enkripsi RC6

3.1.2 *Flowchart* Dekripsi RC6

Pada gambar 2, merupakan *flowchart* yang digunakan untuk dekripsi RC6 dalam penelitian.



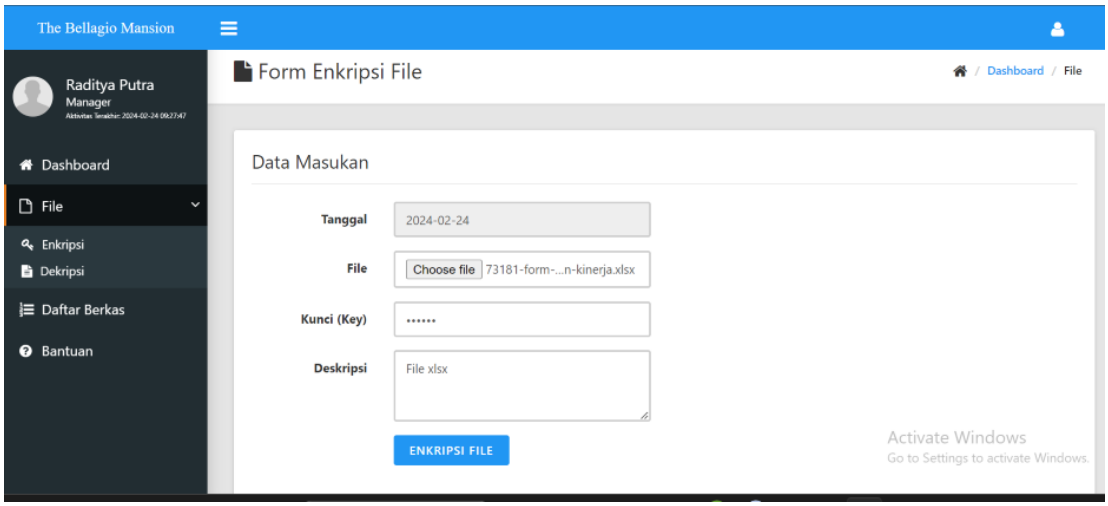
Gambar 2. *Flowchart* Dekripsi RC6

3.2 Implementasi Sistem

Berikut merupakan implementasi pada sistem untuk menjelaskan proses enkripsi dan dekripsi yang terjadi pada algoritme RC6.

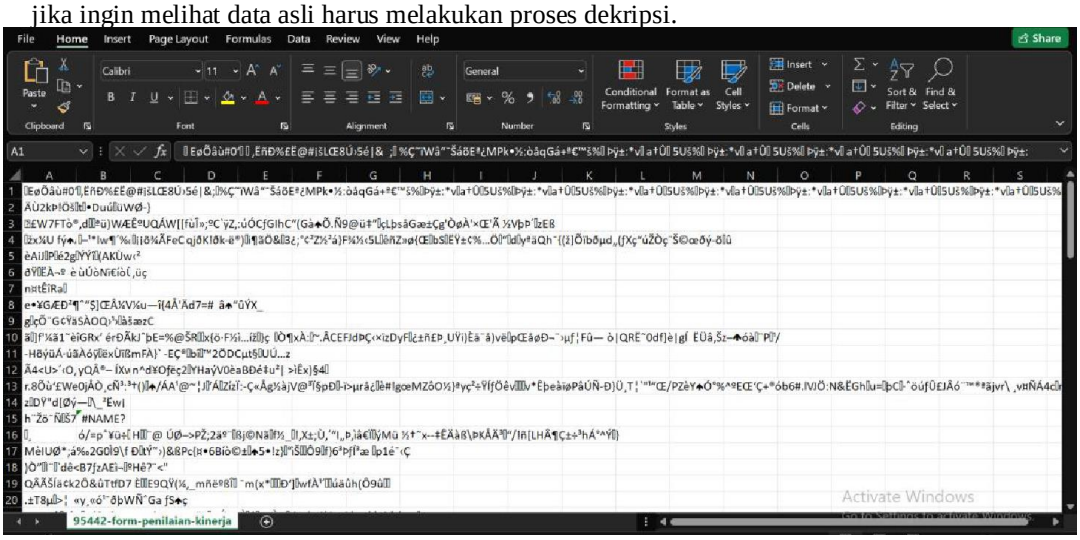
3.2.1 Proses Enkripsi

Pada Gambar 3, diharuskan *login* menggunakan admin, kemudian pilih menu *file* untuk mengenkripsi. Setelah itu, memilih *file* yang ingin dienkripsi yang berekstensi *.pdf, .docx, .xlsx .png, .jpg, .txt dan .pptx*. Jika *file* melebihi 10MB maka aplikasi tidak dapat berjalan. Kemudian memasukkan kunci (*key*) agar dilakukan pembangkitan kunci.



Gambar 3. Proses Enkripsi Pada Sistem

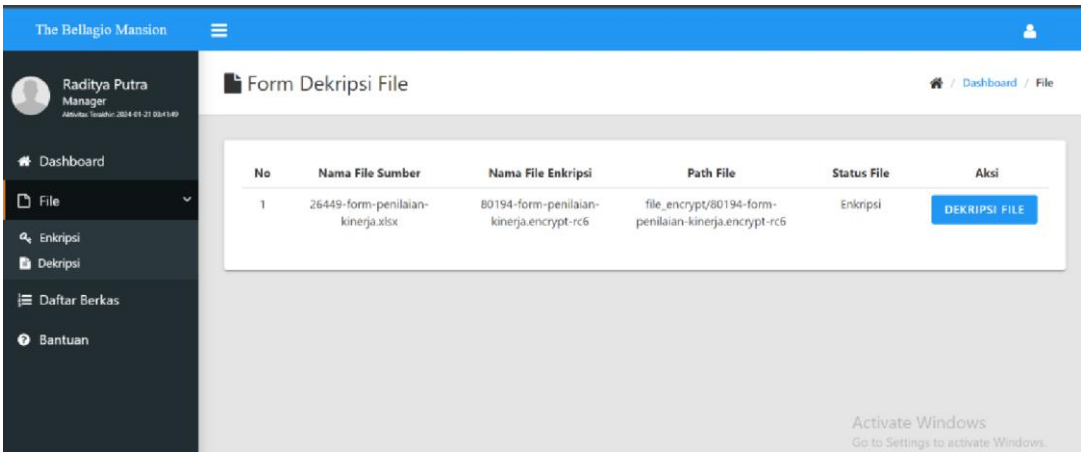
Pada gambar 4, jika *file* tersebut telah terenkripsi yang tadinya formatnya *.xlsx* berubah menjadi format *.encrypt-rc6*. Dan file tersebut tidak akan menampilkan data asli melainkan sudah terenkip dan jika ingin melihat data asli harus melakukan proses dekripsi.



Gambar 4. Hasil Enkripsi

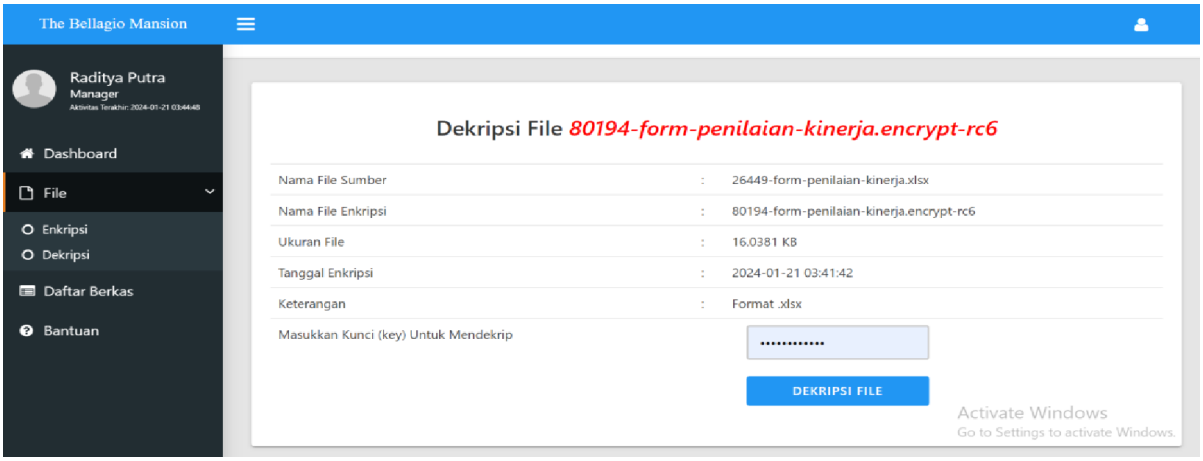
3.2.2 Proses Dekripsi

Pada gambar 5, tahap dekripsi *file* yang statusnya telah terenkripsi, lalu admin akan memilih *file* mana yang akan didekripsi. Jika sudah memilih admin harus menekan tombol dekripsi *file* untuk menuju halaman dekripsi *file*.



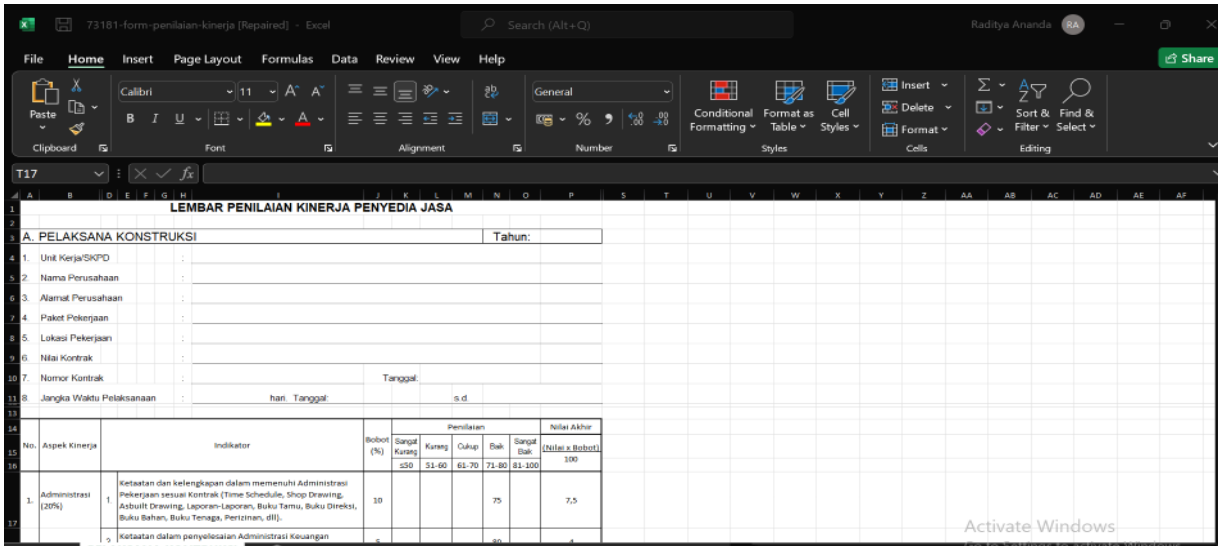
Gambar 5. Daftar *File* Yang Telah Terenkripsi

Pada gambar 6, diarahkan ke halaman dekripsi *file* untuk proses mendekripsikan *file* tersebut agar bisa terbuka kembali dengan memasukkan kunci *(key*) yang sesuai pada saat mengenkripsi *file* tersebut.



Gambar 6. Proses Dekripsi Pada Sistem

Kemudian pada gambar 7, *file* yang sebelumnya terenkripsi sudah bisa dilihat kembali karena telah didekripsi.



Gambar 7. Hasil Dekripsi

3.3 Hasil Pengujian

Pada tabel 1 dan 2, merupakan perbandingan antara pengujian enkripsi dan dekripsi *file*. Pengujiannya terdiri dari *file* ekstensi, besar ukuran pada *file*, waktu untuk melakukan enkripsi, dan waktu untuk melakukan dekripsi.

Tabel 1. Pengujian Pada *File* Enkripsi

Nama *File* Awal	Ekstensi	Ukuran *File* (KB)	Waktu Enkripsi (Seconds)	Ukuran Hasil Enkripsi (KB)	Nama *File* Hasil Enkripsi
List Entries	docx	90	1.77 s	93	34596-list-entries.encrypt-rc6
Company Profile	pptx	2.521	32.46 s	2.581	22327-company-profile.encrypt-rc6
Laporan_Keuangan _Q1_2023	pdf	2.016	25.51 s	2.063	98919-laporan_Keuangan_q1_2023.encrypt-rc6
bellman	jpg	209	3.20 s	212	55845-bellman.encrypt-rc6
bellman	png	429	6.14 s	432	43132-bellman.encrypt-rc6
Registration	txt	14	0.80 s	16	60956-registration.encrypt-rc6

Tabel 2. Pengujian Pada *File* Dekripsi

Nama *File Cipher*	Ekstensi	Ukuran *File* (Kb)	Waktu Dekripsi (Seconds)	Ukuran Hasil Dekripsi(Kb)	Nama *File* Hasil Dekripsi
34596-list-entries.encrypt-rc6	docx	93	1.63 s	90	87941-list-entries
22327-company-profile.encrypt-rc6	pptx	2.581	32.60 s	2.521	30569-company-profile
98919-laporan_Keuangan_q1_2023.encrypt-rc6	pdf	2.063	24.14 s	2.016	3123-laporan_keuangan_q2_2023
55845-bellman.encrypt-rc6	jpg	212	3.07 s	209	38557-bellman
43132-bellman.encrypt-rc6	png	432	4.28 s	429	6442-bellman
60956-registration.encrypt-rc6	txt	16	0.62 s	14	28654-registration

3.4 Analisa Pengujian

Berdasarkan pengujian program enkripsi dan dekripsi yang telah dilakukan, baik itu berupa *file* dokumen docx, pptx, pdf, jpg, png, dan txt . Dari hasil pengujian program yang telah dilakukan, ditemukan beberapa kelebihan dan kekurangan aplikasi ini yaitu:

a. Kecepatan untuk mendekripsi file lebih cepat dibandingkan kecepatan mengenkripsi *file*.
b. Ukuran file menentukan kecepatan enkripsi atau dekripsi. Semakin kecil ukuran file maka semakin cepat juga proses enkripsi maupun dekripsinya
c. Hasil dari output *file* yang telah dienkripsi maupun didekripsi akan memiliki nama *file* yang berbeda.
d. *File* yang telah dienkripsi tidak dapat dibuka atau dikembalikan seperti semula tanpa proses dekripsi.

4. KESIMPULAN

Kesimpulan pada penelitian ini yaitu untuk mengamankan data *file* laporan keuangan yang ada di The Bellagio Mansion yaitu dengan membuat aplikasi kriptografi dengan memasukkan *file* laporan tersebut untuk dienkripsi agar pihak yang tidak berkepentingan tidak bisa membuka *file* perusahaan. Hasil pengujian yang telah dilakukan pada aplikasi yaitu kecepatan untuk mendekripsi file lebih cepat dibandingkan kecepatan mengenkripsi file. Semakin kecil ukuran *file* maka semakin cepat juga proses enkripsi maupun dekripsinya begitupun sebaliknya. Hasil output *file* yang telah dienkripsi maupun yang telah didekripsi memiliki nama *file* yang berbeda. Tanpa proses dekripsi, *file* yang telah dienkripsi tidak dapat dibuka. Beberapa saran untuk penelitian ini yaitu diharapkan dapat mengembangkan program pengamanan *file* ini dengan memodifikasi algoritma pada RC6 atau dengan menambahkan algoritma kriptografi lainnya. Diharapkan juga sistem dapat dibuat dengan versi *mobile* yang dimana fungsinya akan digunakan pada smartphone atau *device mobile* lainnya.

1. DAFTAR PUSTAKA

[1] Ritonga, A.R., D. Irmayani, and A.A. Ritonga, *Rivest Cipher 6 Algorithm Method to secure messages of Medical Record files at Perlayuan Health Center.* Sinkron: jurnal dan penelitian teknik informatika, 2022. 7(2): p. 586-594.
[2] Tena, S., B.J. Mooy, and S.I. Pella, *Implementasi Algoritma Rivest Code 6 (Rc6) Dan Steganografi Least Significant Bit (Lsb) Untuk Keamanan Data Citra Digital.* Jurnal Media Elektro, 2019: p. 107-113.
[3] Berisha, A. and H. Kastrati. *Parallel Implementation of RC6 Algorithm*. 2021.
[4] Kristianto, B.D., G. Gat, and G. Syarifudin, *PERANCANGAN PERANGKAT LUNAK ENKRIPSI SMS MENGGUNAKAN ALGORITMA RC6 DAN RIJNDAEL PADA SMARTPHONE.* SISFOTENIKA, 2020. 10(1): p. 115-126.
[5] Susmitha, C., et al. *Hybrid Cryptography for Secure File Storage.* in *2023 7th International Conference on Computing Methodologies and Communication (ICCMC)*. 2023. IEEE.
[6] Sofyan, M.H. and H. Fahmi, *IMPLEMENTASI ALGORITMA RC6 DALAM PENGAMANAN FILE PADA BP2RD SU SAMSAT SEI RAMPAH.* Jurnal Sistem Informasi Kaputama (JSIK), 2022. 6(1): p. 1-6.
[7] Suzanti, I.O., et al., *Secure Data Flow Messaging on Web Socket using Rivest Code 6.* 2022.
[8] Siswanto, S., et al., *Penerapan Algoritme Kriptografi RC6 Untuk Mengamankan File Penjualan Dan Gambar Produk Alisan.* Prosiding SISFOTEK, 2023. 7(1): p. 30-35.
[9] Yusuf, H., A. Afriyudi, and H. Syaputra, *Digital Signature Pada Citra Digital Menggunakan Algoritma Rc6 Studi Kasus: Dokumen Kartu Keluarga.* Journal of Computer and Information Systems Ampera, 2020. 1(1): p. 44-52.
[10] Seta, H.B., R. Yulistiani, and T. Theresiawati, *PENGAMANAN CITRA DIGITAL REKAM MEDIS MENGGUNAKAN PERPADUAN HASHING ALGORITMA KECCAK DAN RIVEST CODE 6.* Jurnal Ilmiah MATRIK, 2020. 22(3): p. 257-269.