IMPLEMENTASI ENKRIPSI EMAIL STEGANOGRAFI DAN KRIPTOGRAFI MENGGUNAKAN METODE END OF FILE DAN ALGORITMA RC 4 (RIVEST CODE 4) BERBASIS JAVA DESKTOP PADA PT. BIRU SENTRA PERKASA

Ari Komala Sari[1)], Subandi[2)]
Teknik Informatika, Fakultas Teknologi Informasi, Universitas Budi Luhur
Jl. Raya Ciledug, Petukangan Utara, Kebayoran Lama, Jakarta Selatan 12260
E-mail : arikomalasari05@gmail.com[1)], subandionline@gmail.com[2)]

Abstrak

Email adalah salah satu teknologi yang paling disenangi orang saat ini. Dengan menggunakan email kita tidak perlu repot pergi ke kantor pos atau jasa pengiriman lainnya untuk mengirimkan suatu informasi. Email juga memberikan fasilitas yang cukup memadai kita bisa mengirimkan dokumen yang penting melalui internet. Dan kita juga tidak perlu untuk bertemu dengan penerima informasi. Faktor inilah yang menjadi alasan PT. Biru Sentra Perkasa menggunakan email sebagai media pertukaran informasi.Dalam perkembangan teknologi yang sangat pesat saat ini dan diimbangi dengan tingkat kriminal dunia maya atau cybercrime yang merajalela. Tidak jarang akun email dari seseorang menjadi korban dari serangan para Hacker yang melakukan cybercrime, baik itu melalui teknik phising, keylogger, wifisniffing, maupun social engineering untuk mendapatkan password dan segala data yang ada dalam akun tersebut. Dari uraian permasalahan yang telah diuraikan di atas, maka penulis berusaha untuk membuat sebuah aplikasi untuk melakukan pengamanan email seseorang dengan menerapkan algoritma kriptografi RC 4 (Rivest Code 4) dan metode steganografi EOF (End Of File), aplikasi ini dibangun dengan bahasa pemrograman Java berbasis desktop. Aplikasi ini dapat mengamankan dan menjaga kerahasiaan data dan informasi pada PT. Biru Sentra Perkasa dari terjadinya pencurian dan manipulasi data oleh pihak yang tidak berkepentingan.Data yang dapat dienkripsi dan disisipkan dengan format berupa fileword (.docx), file excel (.xlsx), file power point (.pptx), dan file pdf (.pdf) sedangkan untuk file penampung data rahasia berupa file gambar yang berjenis .jpg, .png, .bmp. Dengan menggunakan aplikasi ini, diharapkan pengguna dapat mengirimkan pesan yang sifatnya rahasia karena telah termanipulasi dengan aplikasi tersebut tanpa adanya rasa takut apabila ada seseorang yang tidak bertanggung jawab melakukan pembajakan email.

Kata kunci: Steganografi, *End of File*, Kriptografi, RC 4, Email

1. PENDAHULUAN

seorang tokoh sejarah yang terkenal, Kaisar Romawi, Julius Caesar (100SM – 40SM) yang menggunakan suatu teknik atau metode yang diciptakan dalam menjaga kerahasiaan pesan atau strategi berperang yang akan ia sampaikan kepada prajuritnya yang berada di medan perang. Inilah awal mula kriptografi yang dipergunakan oleh manusia, sehingga menyebabkan semakin kompleksnya perkembangan ilmu kriptografi guna memenuhi tuntutan keamanan dan integritas suatu data atau infromasi terhadap pihak-pihak yang tidak bertanggungjawab. Diperusahaan kontraktor yang bergerak dibidang pembangunan rumah seperti PT. Biru Sentra Perkasa, informasi yang diterima atau dikirim melalui email merupakan salah satu data yang paling penting. Oleh karena itu dibutuhkan suatu metode yang dapat menjaga rahasia informasi tersebut. Metode yang dimaksud adalah kriptografi dan steganografi yang merupakan sebuah seni dan bidang keilmuan dalam penyandian informasi atau pesan dengan tujuan menjaga keamanannya. Walaupun telah berkembang sejak zaman dulu kala, teknik kriptografi yang dibutuhkan masa kini tetap harus menyesuaikan dirinya terhadap meluasnya penggunaan komputer digital pada masa kini.Dalam perkembangan teknologi sekarang ini, teknologi email (Electronic Mail) adalah merupakan suatu teknologi yang sangat diminati dan sangat penting dalam dunia komunikasi. Dengan adanya email, manusia dapat mengirimkan pesan dan juga data melalui teknologi informasi. Pada jaman teknologi yang berkembang pesat, pengiriman surat melalui kantor pos sudah sangat jarang digunakan. Hal ini dikarenakan pengiriman melalui kantor pos membutuhkan waktu yang lama dan proses yang kurang praktis. Faktor inilah yang semakin menguatkan manfaat email dalam kehidupan sehari hari. Pengiriman surat secara tradisional atau melalui kantor pos masih digunakan untuk komunikasi yang berifat formal, seperti antar instansi pemerintah dan dunia pendidikan. Namun, terkadang banyak orang yang beralih ke dalam penggunaan email untuk pengiriman surat formal karena lebih praktis dan cepat. Namun, selain berbagai macam keuntungan dan kemudahan yang diberikan dengan teknologi tersebut, ada kemungkinan potensi bahaya yang akan timbul.Salah satunya adalah terjadinya kebocoran data atau informasi yang ditransmisikan. Kebocoran data ini akan sangat memungkinkan terjadi karena pengiriman email melalui internet akan melalui proses yang lumayan panjang yakni

melewati beberapa server. Dan diperburuk oleh kehadiran hacker yang membobol akun email untuk mendapatkan suatu informasi di dalam akun tersebut. Oleh karena itu, maka perlu dilakukan sebuah pengamanan yakni dengan sebuah penyandian (enkripsi) email untuk melakukan pengacakan pesandan data dalam sebuah email. Pengamanan ini dilakukan untuk menghindari pembacaan oleh orang lain yang tidak bertanggung jawab, kecuali pada orang yang berhak menerimanya.

2. METODE PENELITIAN

2.1 Analisa Masalah

Email adalah salah satu teknologi yang paling disenangi orang saat ini. Dengan menggunakan email kita tidak perlu repot pergi ke kantor pos atau jasa pengiriman lainnya untuk mengirimkan suatu informasi. *Email* juga memberikan fasilitas yang cukup memadai. Kita bisa mengirimkan dokumen yang penting melalui internet. Dan kita juga tidak perlu untuk bertemu dengan Penerima informasi.Faktor inilah yang menjadi alasan PT. Biru Sentra Perkasa menggunakan*email* sebagai media pertukaran informasi.

Dalam perkembangan teknologi yang sangat pesat saat ini dan diimbangi dengan tingkat kriminal dunia maya atau *cybercrime* yang merajalela. Tidak jarang akun *email* dari seseorang menjadi korban dari serangan para *Hacker* yang melakukan *cybercrime***,** baik itu melalui teknik *phising*, *keylogger*, *wifisniffing*, maupun *socialengineering* untuk mendapatkan *password* dan segala data yang ada dalam akun tersebut. Apabila hal itu sudah terjadi, sangat memungkinkan bahwa semua isi yang ada di dalam akun *email* tersebut akan dapat terbaca oleh *Hacker*, apalagi dalam pesan itu terdapat *file* dan pesan yang bersifat rahasia.

2.2 Strategi Penyelesaian Masalah

Untuk memecahkan masalah diatas, maka penulis berusahamembuat sebuah aplikasi untuk melakukan pengamanan *email* seseorang dengan menerapkan ilmu kriptografi dan steganografi.Sehingga isi dari dokumen tersebut tidak dicurigai dan tidak bisa dibaca atau tidak bisa diketahui oleh pihak lain yang tidak berhak atas dokumen tersebut. Dengan kriptografi aplikasi tersebut akan mengacak sebuah data menjadi data yang isinya tidak bisa dibaca. Agar data tersebut lebih terjaga kerahasiaannya, melalui metode steganografi *file* tersebut akan disembunyikan ke dalam gambar agar tidak dicurigai sebagai data rahasia. Kemudian mengembalikan data rahasia tersebut menjadi seperti semula tanpa mengalami perubahan sedikitpun.

Dari berbagai macam jenis kriptografi yang dianalisa dari media internet, buku, jurnal dan lainnya untuk mengetahui algoritma manakah yang banyak dibicarakan dalam maslah kekuatan kunci untuk mengamankan sebuah data, pada akhirnya penulis memilih algoritma RC 4 (*Rivest Code 4)* sebagai metode kriptografi dan Algoritma EOF pada metode steganografi. RC4 termasuk ke dalam *cipher* aliran (*stream cipher*) dengan kunci privat / kunci simetri (kunci yang sama digunakan untuk proses enkripsi dan dekripsi), sedangkan EOF adalah salah satu metode steganografi dimana penyisipan data dimulai dengan menambahkan *bit* penampung pada gambar kemudian diberi *header* atau *end* sebagai penanda. Pertama *file* akan dienkripsi menjadi *file Chiphertext,* kemudian hasil enkripsi akan disembunyikan dengan cara disisipkan ke dalam media gambar (*covertext)* dan mengembalikannya menjadi *file* semula pada proses dekripsi.

2.3 Rancangan Program

Program yang dibuat terdiri dari beberapa *Form*, yang terdiri dari *Form Login*, *Form* Menu Utama, *Form* Embed & Enkripsi Dokumen,*Form Retrieve* & Deskripsi Dokumen,*Form Inbox Mail*, *Form* Baca *Email Inbox*, *Form* Tulis *Email*, *Form* Balas *Email*, dan *Form* Teruskan *Email* . Pada *Form Menu* Utama terdapat dua buah *menu* yang dapat dijalankan sesuai dengan kebutuhan pengguna, yaitu *Menu User* yang memiliki *submenu Logout* dan *Menu* Amankan Dokumen yang memiliki *submenu Form MenuEmbed*& Enkripsi Dokumen, *Form Menu Retrieve* & Dekripsi Dokumen.

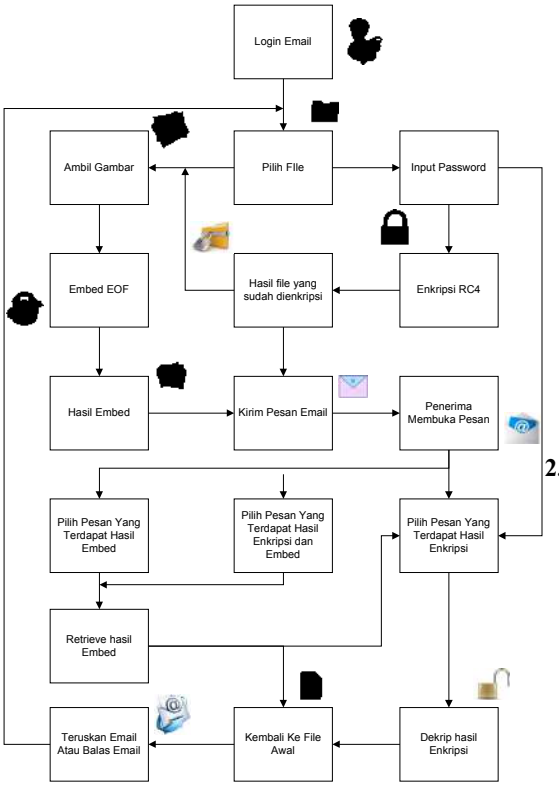
Submenu Logout bertujuan untuk keluar dari akun *email*. *SubmenuFormEmbed*& Enkripsi Dokumen bertujuan untuk melakukan pengenkripsian dokumen beserta penyisipan dokumen ke dalam gambar, Pengguna terlebih dahulu diharuskan memilih *file* yang berekstensi.*docx, .pptx, .xlsx*, *.pdf*, dan penggguna harus memilih gambar untuk disisipkan dokumen kemudian memasukkan kata sandi untuk melakukan pengenkripsian *file*, setelah itu barulah pengguna memilih tombol amankan dokumen untuk mendapatkan hasil gambar yang sudah disisipkan dokumen yang telah di enkrip. *Submenu Form Retrieve &* Dekripsi Dokumen digunakan untuk mengembalikan *file* yang sudah disisipkan dan mendekripsi dokumen hasil enkripsi. Dengan cara pengguna harus mempunyai dokumen hasil *embed*& enkripsi yang berupa gambar dan program akan otomatis melakukan *retrieve* pada gambar untuk mendekripsi dokumen yang telah disisipkan dengan *password*nya, pengguna juga dapat mengenkripsi saja serta mendekripsi dokumen yang terenkripsi beserta *password* nya.

2.4 Skema Proses Aplikasi

Memerlukan *input*an berupa *file, password*, dan gambar. *User* harus *login* dengan *email* sendiri terlebih dahulu. Pilih Tulis *Email* untuk mengirim pesan. Lalu *user* memasukkan *email* penerima dan subject beserta isi pesan. Setelah itu *user* melampirkan dokumen, ketika user melampirkan

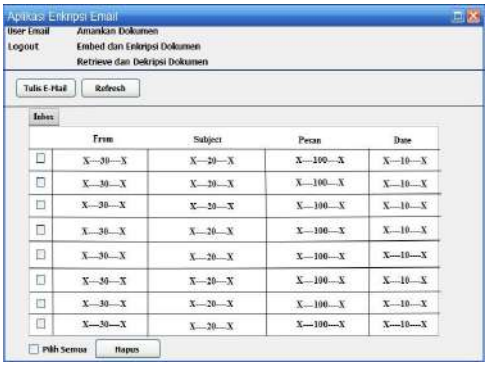
dokumen akan muncul *form* untuk *embed* dan enkripsi, proses pertama yang dilakukan adalah *user* harus memilih *file* untuk di enkripsi kemudian ceklis proses enkripsi. Setelah itu user memilih *file* berupa gambar. Lalu *user* memasukan *password* untuk enkripsi. *File* yang telah terenkrip akan disisipkan ke dalam sebuah gambar. Gambar yang ter*embed*akan ada penanda pesan pada akhir byte gambar. Jika penerima mempunyai aplikasi ini, penerima dapat *retrieve* dan dekripsi dokumen. Dengan cara membuka pesan yang terdapat lampiran dokumen berupa gambar. Lalu penerima menyimpan *file*, jika *file* tersisipi otomatis program akan membaca dokumen apabila tersisipi *file* lain, penerima harus memasukan *password* apabila terdapat enkripsi disebuah *file*. *File* yang telah terenkrip dan terembed akan kembali seperti *file* awal Dan pilih teruskan jika ingin meneruskan pesan, atau pilih balas untuk membalas pesan. Jika diperlukan pengenkripsian atau pengdekripsian file kembali kita dapat mengulangi langkah.

Skema proses keseluruhan aplikasi dapat dilihat pada gambar 1 berikut dibawah ini :



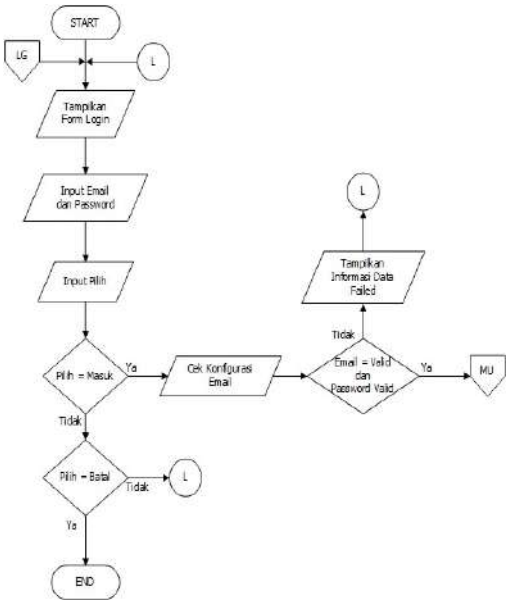
Gambar 1 : Rich Picture Proses Aplikasi

2.5 Rancangan Layar Menu utama



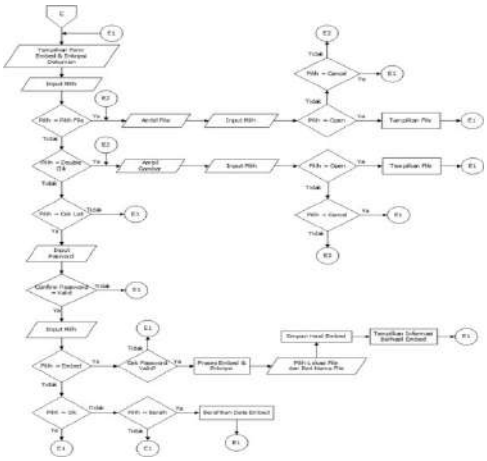
Gambar 2 : Rancangan Layar Form Menu Utama

2.6 Flowchart Form Login



Gambar 3 : Flowchart Form Login

2.7 Flowchart Form Menu Enkripsi Dokumen



Gambar 4 : Flowhcart Form Menu Enkripsi Dokumen

3. HASIL DAN PEMBAHASAN

3.1 Perangkat Yang Digunakan

Pada tahap kebutuhan perangkat ini dilakukan pengumpulan kebutuhan – kebutuhan dari semua elemen sistem perangkat yang akan digunakan dalam pembuatan program. Adapun perangkat keras dan perangkat lunak yang digunakan antara lain sebagai berikut :

a. Perangkat Keras (Hardware)

Perangkat keras (*hardware*) yang digunakan selama tahap pengimplementasian aplikasi ini, antara lain:

1) Processor : Intel Core i3
2) RAM : 4.00 GB
3) Hard Disk : 320 GB

b. Perangkat Lunak (Software)

Berikut adalah perangkat lunak (*software*) yang digunakan selama tahap pengimplementasian aplikasi ini, diantaranya:

1) Sistem Operasi *MicrosoftWindows*7
2) *Netbeans* IDE 7.4
3) JDK 7.0.4

3.2 Penggunaan Aplikasi

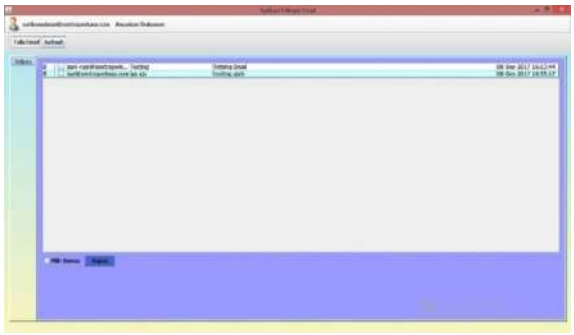
a. Form Login

Ketika kita menjalankan aplikasi yang pertama kali tampil adalah *form login*. Jadi pengguna terlebih dahulu diminta untuk memasukkan *email* dan *password* secara benar agar dapat masuk ke dalam aplikasi ini dan menggunakan aplikasi ini sesuai dengan keinginan pengguna. Berdasarkan rancangan layar yang telah digambarkan sebelumnya, maka bentuk tampilan aplikasi ketika berada pada *form login* seperti gambar dibawah ini.

Gambar 5 : Tampilan Form Login

b. Form Menu Utama

Menu form *men*u utama akan muncul saat tombol menu data dan memilih menu *dataset* pada Menu Utama. Form *menu utama* dapat dilihat pada gambar 6 berikut ini :



Gambar 6 : Tampilan Form Menu Utama

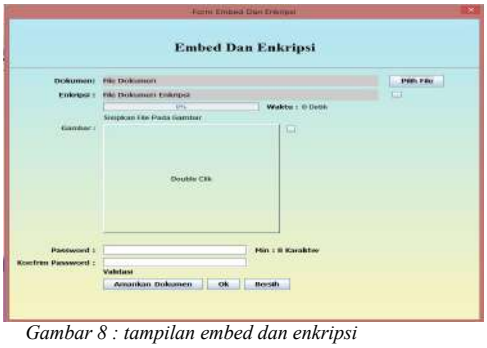
c. Tampilan Layar Form Tulis Email

Menu form Tulis Email ini akan muncul saat tombol menu tulis pesan pada menu utama dilihat pada gambar 7 berikut ini :

Gambar 7 : Tampilan Form Tulis Email

d. Tampilan Embed dan Enkripsi Dokumen

Tampilan *form* ini akan tampil ketika *user* melampirkan data pada Tulis *Email* dan dapat di akses juga melalui *menu* amankan dokumen dengan memilih *submenuEmbed* Enkripsi Dokumen. *Form* tersebut berfungsi untuk mengenkripsi data dan meng-*embed* data yang sudah di enkripsi ataupun tidak di enkripsi. Tampilan *form* akan tampil seperti gambar berikut ini

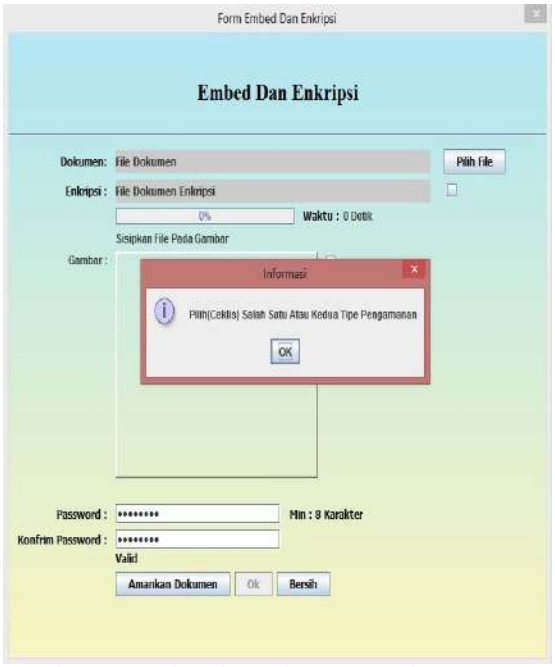


Gambar 8 : tampilan embed dan enkripsi

Apabila *user* mengklik amankan dokumen saat *form* kosong, akan tampil *form* informasi seperti pada gambar berikut ini.

Gambar 9 : Password Tidak valid

Kemudian apabila *password* di *input*kan dengan dokumen kosong maka akan keluar informasi seperti gambar berikut ini.

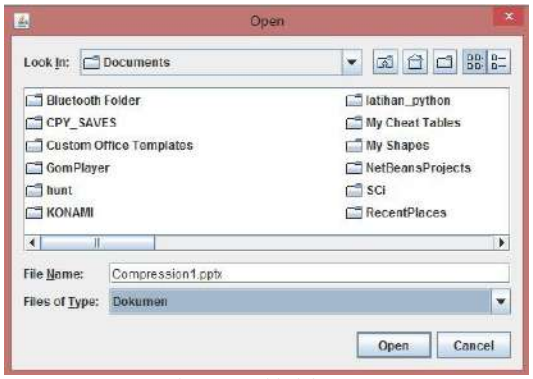


Gambar 10 : tampilan informasi hanya password

Apabila *user* telah menceklis tipe pengamanan dengan dokumen kosong maka akan tampil informasi seperti gambar berikut ini.

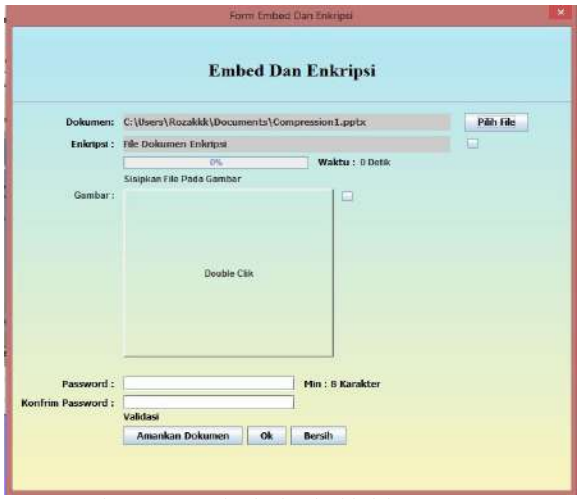
Gambar 11 : Tampilan belum memilih dokumen

Ketika *user* memilih *button* pilih *file* maka akan keluar *form* pilih dokumen seperti gambar berikut ini.



Gambar 12 : Pilih dokumen

Ketika *user* telah berhasil memilih dokumen maka *file* akan tampil di *list* dokumen seperti gambar berikut ini.



Gambar 13 : Tampilan berhasil pilih dokumen

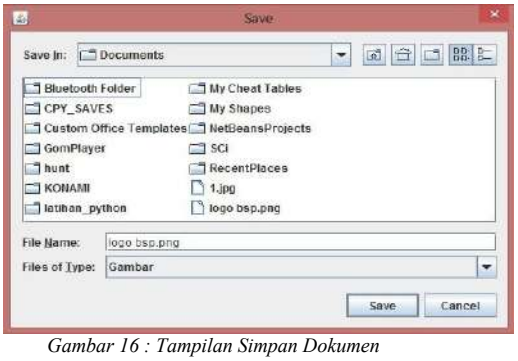
memilih maka gambar akan tampil di *list* gambar seperti gambar berikut ini.

Gambar 14 : tampilan memilih gambar

Jika *passwordvalid*, maka akan tampil informasi *valid*. Kemudian *user* dapat melanjutkan proses enkripsi dan *embed* dengan mengklik amankan dokumen, maka proses akan berjalan seperti gambar berikut ini.

Gambar 15 : Tampilan amankan proses enkripsi

Bila proses enkripsi dan *Embed* telah berhasil, maka akan tampil *form* untuk menyimpan hasil *file* enkripsi dan *embed* tersebut seperti gambar berikut ini.



Gambar 16 : Tampilan Simpan Dokumen

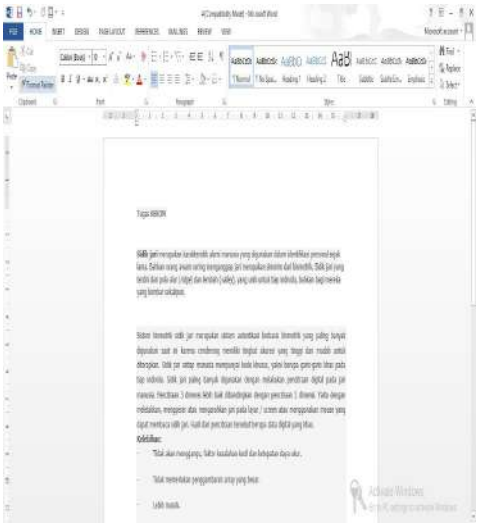
Bila penyimpanan *file* berhasil, maka akan tampil *form* informasi enkrip *Embed* berhasil seperti gambar berikut ini.

Gambar17 :Tampilan Enkrip Embed Berhasil

3.3 Pengujian Aplikasi

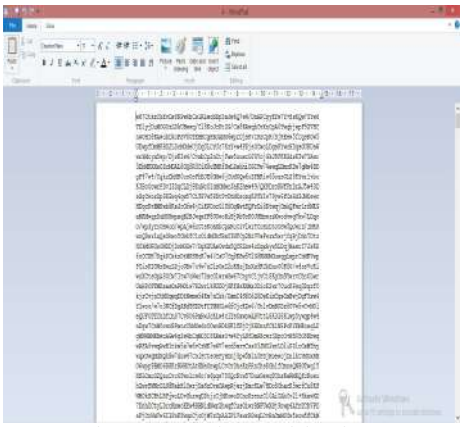
Dalam melakukan pengujian penulis memanfaatkan beberapa berkas *file* dokumen sebagai data yang akan disisipkan dalam *file*. Berkas – berkas yang akan digunakan tersebut adalah sebagai berikut.

a. Proses enkripsi dokumen file *.doc



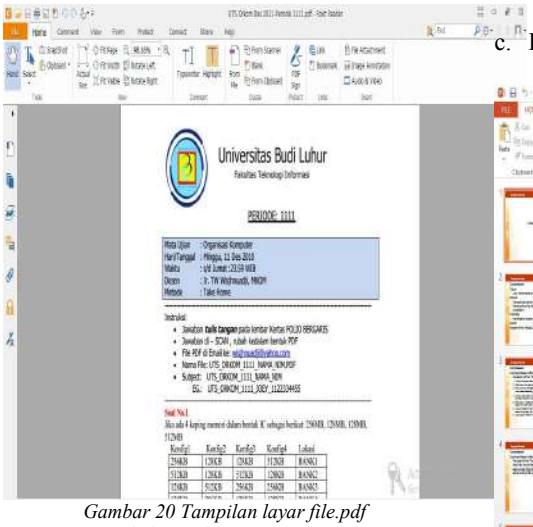
*Gambar 18 :**Tampilan Layar File .docx SebelumProsesEnkripsi***

Setelah dilakukan proses enkripsi maka *file* akan menjadi teracak dan sulit dipahami, *file .docx* yang telah dienkripsi tersebut dibuka dengan menggunakan *wordpad*. Hasil nya dapat dilihat seperti gambar berikut ini.



Gambar1 9 : Tampilan Sesudah Enkripsi

b. Proses Enkripsi dokumen file *.pdf

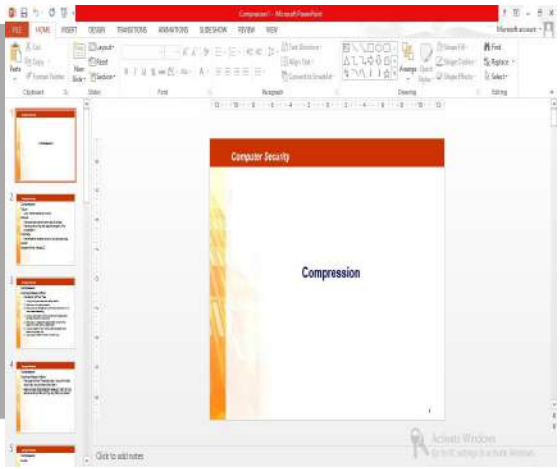


Gambar 20 Tampilan layar file.pdf sebelum enkripsi

Setelah dilakukan proses enkripsi maka *file* akan menjadi teracak dan sulit dipahami, *file .pdf* yang telah dienkripsi tersebut dibuka dengan menggunakan *wordpad*. Hasil nya dapat dilihat seperti gambar berikut ini.

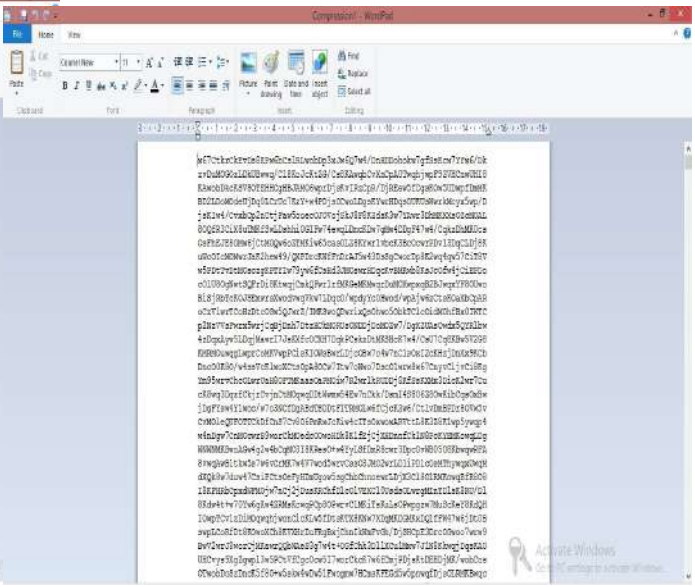
Gambar 21: Tampilan selesai enkripsi

c. Proses Enkripsi dokumen *.pptx



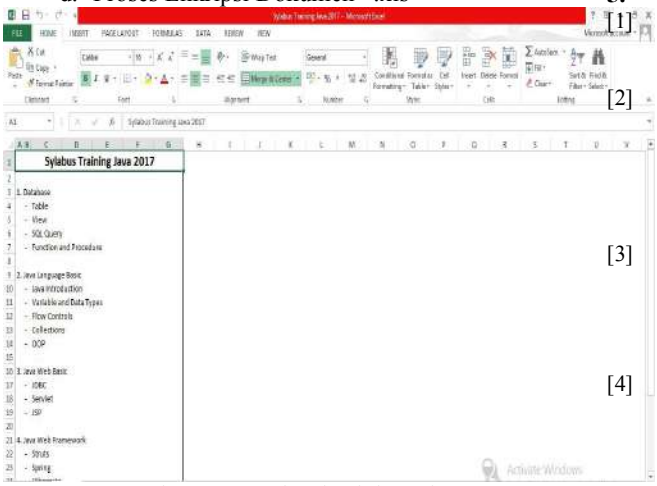
Gambar 22 : Tampilan .pptx sebelum enkripsi

Setelah dilakukan proses enkripsi maka *file* akan menjadi teracak dan sulit dipahami, *file .pptx* yang telah dienkripsi tersebut dibuka dengan menggunakan *wordpad*. Hasil nya dapat dilihat seperti gambar berikut ini.



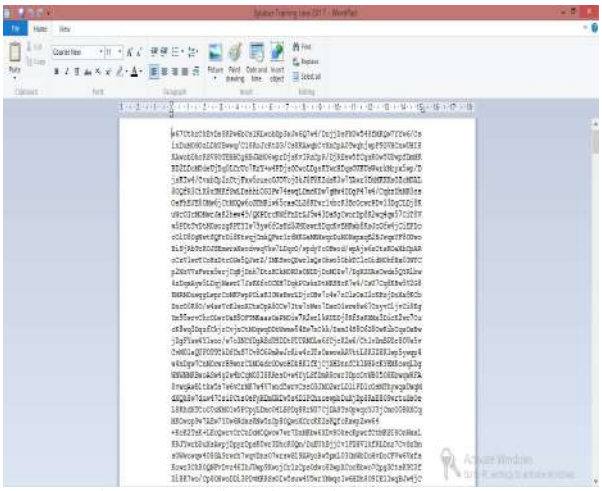
Gambar 23 : Tampilan Selesai enkripsi

d. Proses Enkripsi Dokumen *.xls



Gambar 24 : Tampilan xls sebelum enkripsi

Setelah dilakukan proses enkripsi maka *file* akan menjadi teracak dan sulit dipahami, *file .xlsx* yang telah dienkripsi tersebut dibuka dengan menggunakan *wordpad*. Hasil nya dapat dilihat seperti gambar berikut ini.



Gambar 25 : Tampilan Selesai enkripsi

4. KESIMPULAN

Algoritma kriptografi RC 4 dan metode steganografi EOF dapat diimplementasikan pada aplikasi pengamanan dokumen melalui jalur *email*. Dengan adanya program aplikasi kriptografi dan steganografi pengamanan dokumen, penyimpanan dan pertukaran informasi menjadi lebih aman. Waktu yang digunakan untuk mengembed dan me-*retrievefile* berbanding lurus, jika ukuran *file* yang di proses semakin besar maka waktu yang digunakan semakin lama, sedangkan jika ukuran *file* yang di proses semakin kecil maka waktu yang digunakan semakin cepat.

5. DAFTAR PUSTAKA

[1] Y. Ariyanto, "Algoritma RC4 Dalam Proteksi Transmisi Dan Hasil Query Untuk ORDBMS POSTGRESQL," vol. 10, pp. 53–59, 2009.

[2] Nurhadian and A. Pudoli, "Implementasi Keamanan File dengan Kompresi Huffman dan Kriptografi menggunakan Algoritma RC4 serta Steganografi menggunakan End of File Berbasis Desktop pada SMK Negeri 3 Kota Tangerang," *J. TICOM*, vol. 5, no. 1, pp. 39–46, 2016.

[3] A. D. Hidayat and I. Afrianto, "Sistem Kriptografi Citra Digital Pada Jaringan Intranet Menggunakan Metode Kombinasi Chaos Map Dan Teknik Selektif," vol. IX, no. 1, pp. 59–66, 2017.

[4] N. Sumiaty and N. Sumiaty, "Literasi internet pada siswa sekolah menengah pertama," *Penelit. Komun.*, vol. 17, no. 88, 2014.