KOMBINASI ALGORITMA KRIPTOGARAFI AES DAN DES UNTUK ENKRIPSI FILE DOKUMEN PROPOSAL

Candra Irawan[1], Agus Winarno[2]
[1,2]Program Studi Sistem Informasi, Fakultas Ilmu Komputer, Universitas Dian Nuswantoro
e-mail: [1]candra.irawan @dsn.dinus.ac.id, [2]agus.winarno@dsn.dinus.ac.id

Abstrak

DES (Data Encryption Standard) dan AES (Advanced Encryption Standard) keduanya adalah cipher blok simetris. AES diperkenalkan untuk mengatasi kelemahan DES. Karena DES memiliki ukuran kunci yang lebih kecil yang membuatnya kurang aman untuk mengatasi DES tiga kali ini diperkenalkan tetapi ternyata lebih lambat. Perbedaan mendasar antara DES dan AES adalah bahwa dalam blok DES plaintext dibagi menjadi dua bagian sebelum algoritma utama dimulai sedangkan, di AES seluruh blok diproses untuk mendapatkan ciphertext. Dalam makalah ini, telah dibuat sebuah aplikasi kriptosistem dan telah dilakukan uji coba terhadap file dengan beberapa ukuran berbeda. Hasil implementasi dengan filer berisi karakter-karakter acak dengan ukuran file masing-masing 1 MB, 204 KB dan 159 KB. File dokumen yang melaui proses enkripsi dan dekripsi akan mengalami perubahan besar file sebesar 0.05%, dikarenakan mengalami penambahan bit melalui proses enkripsi yang cukup panjang. Dari pengujian Avalanche Effect *kombinasi algoritma aes dan des dapat dikatakan aman dengan persentase 46.38%.*

***Kata Kunci**: Kripto, AES, DES*

1. PENDAHULUAN

Semakin berkembang pesat teknologi sekarang ini, seluruh aktivitas manusia yang dilakukan sangat bergantung pada kemajuan teknologi. Segala sektor kehidupan modern manusia dilakukan secara digital. Sehingga berpengaruh terhadap pertumbuhan data dan informasi dari waktu ke waktu semakin besar dan terus menunjukkan peningkatan. Semakin meningkatnya kebutuhan akan penggunaan data dan informasi maka dibutuhkan perangkat penyimpanan dengan kapasitas yang besar, namun jika penyimpanan sudah tidak mampu untuk menampung lebih banyak data maka diperlukan penghapusan data, untuk mengatasi permasalahan tersebut diperlukan kompresi file menjadi lebih kecil.

Penggunaan internet juga sangat memengaruhi penanganan baik berupa data maupun informasi yang lebih baik salah satunya segi keamanan. maka dari itu perlu diperhatikan apabila terjadi pencurian data maupun penyalahgunaan data dan informasi yang ada oleh orang yang tidak berhak dan tak bertanggung jawab. Dalam hal ini guna mencegah dan mengamankan file – file penting maka diperlukan Kriptografi. Kriptografi dapat dijadikan solusi yang digunakan dalam pengamanan data yang bersifat rahasia. Data yang terkandung dalam sebuah file disandikan dan diubah kebentuk karakter lain secara acak sehingga hanya orang tertentu yang memiliki otoritas yang dapat mengetahui isi dari data yang telah disandikan.

Kriptografi merupakan ilmu yang mempelajari temtang proses pengamanan data. Terdapat 2 jenis kriptografi, yaitu kriptografi klasik dan kriptografi modern. Kriptografi klasik [1] merupakan sebuah proses penyembunyian data menggunakan satu kunci dalam pengamanan data, terdapat 2 tehnik dasar yang digunakan dalam kriptografi klasik yaitu subsitusi dan transposisi. Sedangkan kriptografi modern adalah sebuah algoritma yang kompleks, hal ini disebabkan algoritma modern menggunakan komputer. Berdasarkan permasalahan tersebut, diperlukan suatu analisa dan pengembangan terhadap suatu proses pengamanan data. Dengan membangun system yang bertujuan untuk mengamankan data pada sebuah aplikasi.

Pada dasarnya untuk mengurangi kelemahan dalam pengamanan data yang terdapat pengiriman file yaitu dengan menggunakan algoritma kriptografi yang berfungsi untuk merusak / menyembunyikan data sehingga file yang akan dikirim ke penerima harus di enkripsi terlebih dahulu dengan menggunakan algoritma kriptografi misalnya dengan menggunakan algoritma AES, DES [2] [3], RSA [3], Rijndael, Block Cipher [4] dan lain sebagainya. Pada proses pengamanan data dalam aplikasi ini penulis menggunakan algoritma klasik atau sering disebut dengan algoritma konvensional, adalah algoritma yang dalam proses penyembunyian data menggunakan kunci yang sama. Algoritma kriptografi AES memiliki variasi ukuran block, yaitu 128 bit, 192 bit, dan 256 bit. Pemerintahan *United States* telah menetapkan Algoritma kriptografi AES sebagai standart enkripsi. Standar dalam AES terdiri dari 3 kunci block, yakni AES – 128, AES – 192, dan AES – 256 yang mengadopsi dari korelasi yang pada awalnya dikenal dengan nama Rijndael. M. Yuli Andri telah meneliti mengenai implementasi algoritma kriptografi DES pada berkas digital [5]. Irjatul Wardah telah meneliti mengenai kriptografi algoritma DES yang digunakan untuk pengiriman gambar menggunakan telephone seluler [6]. Indra Syahputra telah meneliti mengenai simulasi keamanan informasi menggunakan kriptografi algoritma DES [7].

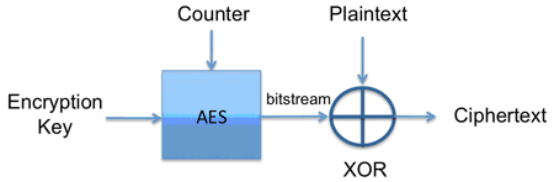
2. TINJAUAN PUSTAKA

2.1 Kriptografi

Kriptografi adalah ilmu mengenai teknik enkripsi dimana "naskah asli" (plaintext) diacak menggunakan suatu kunci enkripsi menjadi "naskah acak yang sulit dibaca" (ciphertext) oleh seseorang yang tidak memiliki kunci dekripsi**.** Kata kriptografi berasal dari bahasa Yunani, *"kryptós"* yang berarti tersembunyi dan *"gráphein"* yang berarti tulisan [8]. Sehingga kata kriptografi dapat diartikan berupa frase "tulisan tersembunyi". Menurut *Request for Comments* (RFC), kriptografi merupakan ilmu matematika yang berhubungan dengan transformasi data untuk membuat artinya tidak dapat dipahami (untuk menyembunyikan maknanya), mencegahnya dari perubahan tanpa izin, atau mencegahnya dari penggunaan yang tidak sah [9]. Jika transformasinya dapat dikembalikan, kriptografi juga bisa diartikan sebagai proses mengubah kembali data yang terenkripsi menjadi bentuk yang dapat dipahami. Artinya, kriptografi dapat diartikan sebagai proses untuk melindungi data dalam arti yang luas. Kriptografi merupakan sebuah ilmu yang digunakan untuk menjaga kemanan pesan. Menurut istilah "seni" didalam kriptografi berasal dari fakta sejarah pada masa awal sejarah kriptografi yang memiliki teknik unik untuk merahasiakan setiap pesan [10]. Teknik merahasiakan pesan didalam kriptografi memiliki estetika tersendiri dalam setiap penulisan pesan rahasia.

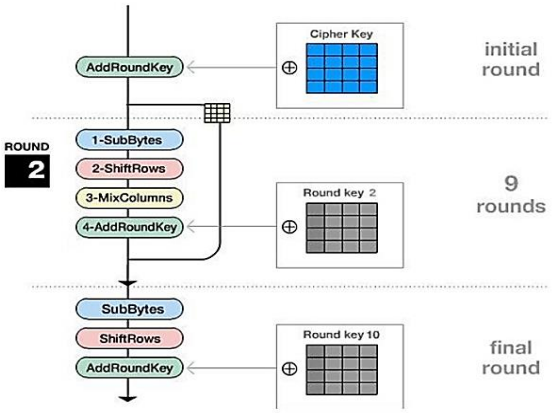
2.2 AES

Algoritma AES (*Advance Encryption Standard*) merupakan algoritma standart enkripsi menggunakan kunci simetris yang diadopsi oleh pemerintahan Amerika Serikat. Setiap algoritma mempunyai ukuran 128-bit menggunakan ukuran *key* masing – masing 128, 192, dan 256 bit [10] seperti ditunjukkan Gambar 1.



Gambar 1. Proses Kriptografi AES *(Advance Encryption Standard)* [11]

Kinerja AES dapat dikatakan sangat baik dikarenakan terdapat metode enkripsi yang bekerja dari beberapa network [12].

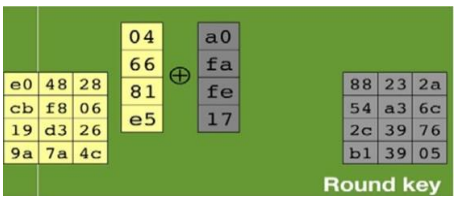


Gambar 2. Diagram Algoritma AES [11]

Berikut ini adalah tahapan proses algoritma algoritma AES seperti yang telah diilustrasikan pada gambar 3.5 diatas :

a. *Add Round Key*

Add Rount Key merupakan sebuah *ciphertext* yang dikombinasikan dalam perhitungannya dengan menggunakan XOR.



Gambar 3. Add Round Key

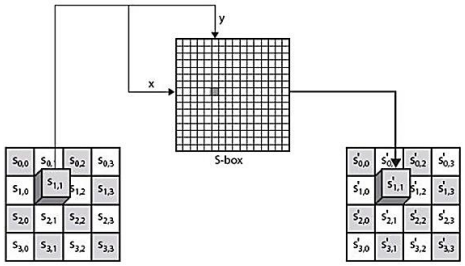
Dalam gambar tersebut terlihat tebel yang berada kiri merupakan perhitungan *ciphertext* dan yang berada disebelah kanan merupakan hasil dari roundkey-nya. XOR didalam yang dilakukan dalam gambar diatas dimana setiap kolom yaitu *ciphertext* pada kolom 1 dilakukan XOR dengan kolom 1 *round key* dan selanjutnya.

b. *Sub Bytes*

Prinsip dasar dalam perhitungan *sub bytes* yaitu mengubah dari isi tabel atau isi matrik lainnya yang disebut dengan S-BOX.

	x0	x1	x2	x3	x4	x5	x6	x7	x8	x9	xa	xb	xc	xd	xe	xf
0x	63	7c	77	7b	f2	6b	6f	c5	30	01	67	2b	fe	d7	ab	76
1x	ca	82	c9	7d	fa	59	47	f0	ad	d4	a2	af	9c	a4	72	c0
2x	b7	fd	93	26	36	3f	f7	cc	34	a5	e5	f1	71	d8	31	15
3x	04	c7	23	c3	18	96	05	9a	07	12	80	e2	eb	27	b2	75
4x	09	83	2c	1a	1b	6e	5a	a0	52	3b	d6	b3	29	e3	2f	84
5x	53	d1	00	ed	20	fc	b1	5b	6a	cb	be	39	4a	4c	58	cf
6x	d0	ef	aa	fb	43	4d	33	85	45	f9	02	7f	50	3c	9f	a8
7x	51	a3	40	8f	92	9d	38	f5	bc	b6	da	21	10	ff	f3	d2
8x	cd	0c	13	ec	5f	97	44	17	c4	a7	7e	3d	64	5d	19	73
9x	60	81	4f	dc	22	2a	90	88	46	ee	b8	14	de	5e	0b	db
ax	e0	32	3a	0a	49	06	24	5c	c2	d3	ac	62	91	95	e4	79
bx	e7	c8	37	6d	8d	d5	4e	a9	6c	56	f4	ea	65	7a	ae	08
cx	ba	78	25	2e	1c	a6	b4	c6	e8	dd	74	1f	4b	bd	8b	8a
dx	70	3e	b5	66	48	03	f6	0e	61	35	57	b9	86	c1	1d	9e
ex	e1	f8	98	11	69	d9	8e	94	9b	1e	87	e9	ce	55	28	df
fx	8c	a1	89	0d	bf	e6	42	68	41	99	2d	0f	b0	54	bb	16

(a)



(b)

Gmabar 4. (a) Tabel S-BOX, (b) Ilustasi *Sub-bytes*

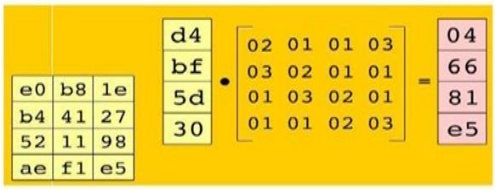
Berdasarkan penggambaran *sub-bytes* diatas terdapat nomor kolom dan baris. Pada tiap-tiap kotak didalam *blok cipher* terdapat informasi dalam bentuk dua digit bilangan hexadecimal, dapat berupa angka, angka huruf, maupun huruf angka yang telah tercantum didalam rijndael S-BOX. Dalam setiap tahapnya diambil satu dari isi kotak pada matrik untuk dicocokkan dengan digit kiri untuk baris dan digit sebelah kanan untuk kolom. Setelah diketahui kolom dan baris mada dapat mengambil isi tabel dari rijndael S-BOX. Langkah terakhir didapatkan blok baru yang berasal dari pengubahan keseluruhan blok cipher yang berisi hasil dari pengukaran tiap – tiap isi pada blok yang telah disebutkan pada langkah sebelumnya.

c. *ShiftRows*

Merupakan pergesaran tiap – tiap elemen blok yang dilakukan perbaris. Pada baris pertama tidak dilakukan pergeseran, baris kedua dilakukan pergesaran sebanyak satu bit lalu baris ketiga dilakukan pergeseran sebanyak tiga bit.

d. *Mix Columns*

Berguna untuk mengalikan setiap elemen pada blok cipher dengan matriks, perkalian dilakukan dengan menggunakan perkalian matriks biasa yang akan dimasukkan ke blok cipher baru, gambaran berikut menjelaskan proses perkalian.



Gambar 5. *Mix Column*

2.3 DES (Data Encryption Standart)

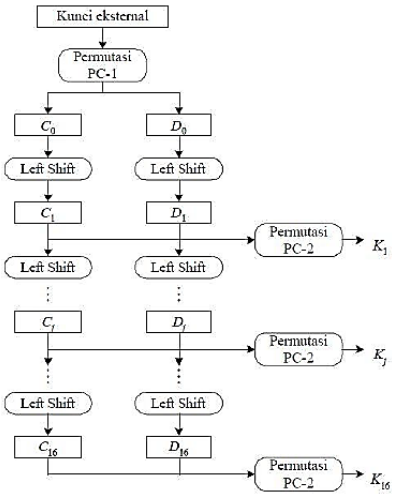
Algoritma DES merupakan algoritma kriptografi yang masuk kedalam jenis simetri dan tergolong jenis blok code. Dalam proses enkripsi algoritma DES mengubah text asli yang berukuran 64 bit menjadi 63 bit text kode dengan 56 bit kunci internal. Pembangkitan kunci dilakukan dari kunci external menjadi kunci internal yang memiliki panjang 64 bit. Skema algoritma DES [5] dapat digambarkan sebagai berikut :

a. *Plaintext* (text asli) diubah menggunakan tabel IP (initial permutation). Dapat dituliskan sebagai berikut x_0 = IP(x) = L_0R_0, dimana L_0 merupakan 32 bit pertama dari x_0 dan 32 bit lainnya merupakan R_0.
b. Kemudian hasil dari IP diputar (*enchipering*) sebanyak 16 putaran. Dimana setiap *enchipering* menggunakan *key* (kunci internal) yang berubah - ubah berdasarkan perhitungan LiRi.

   $1 \leq i \leq 16$, dengan mengikuti aturan berikut:

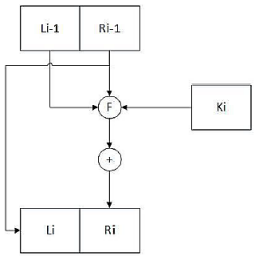
   Li = Ri – 1

   Ri = Li - 1 XOR f (Ri – 1, Ki) (1)



Gambar 6. Permutasi Kunci

Dimana XOR adalah *exclusive-or* dari dua F. Dimana F merupakan suatu fungsi dan K1 hingga K16 yang memiliki panjang 48 yang menggunakan perhitungan fungsi dari kunci K. Putaran (*enchipering*) dari proses enkripsi dapat ditunjukkan berdasarkan gambar dibawah ini.



Gambar 7. Skema satu putaran DES

c. Hasil dari proses tersebut kemudian dilakukan invers initial permutation (IP-1) menggunakan tabel IP-1 sehingga menjadi blok teks kode. IP-1 ke bit string R16L16, memperoleh teks-kode y, kemudian dirumuskan menggunakan y=IP-1(R16L16).

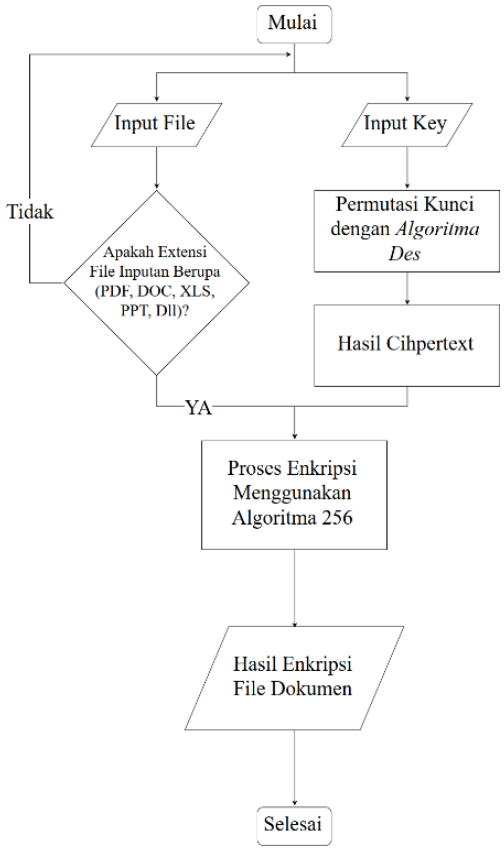
Dalam proses enciphering, plaintext terbagi menjadi dua bagian, kiri (L) dan kanan (R), yang masing-masing panjangnya 32 bit. Pada setiap putaran i, blok R merupakan masukan untuk fungsi transformasi yang disebut f. Pada fungsi f, blok R dikombinasikan dengan kunci internal Ki. Keluaran dari fungsi f di-XOR-kan dengan blok *mL* untuk mendapatkan blok *Rm* yang baru. Sedangkan blok L yang baru diambil dari blok R sebelumnya. Ini adalah satu putaran DES.

3. METODE PENELITIAN

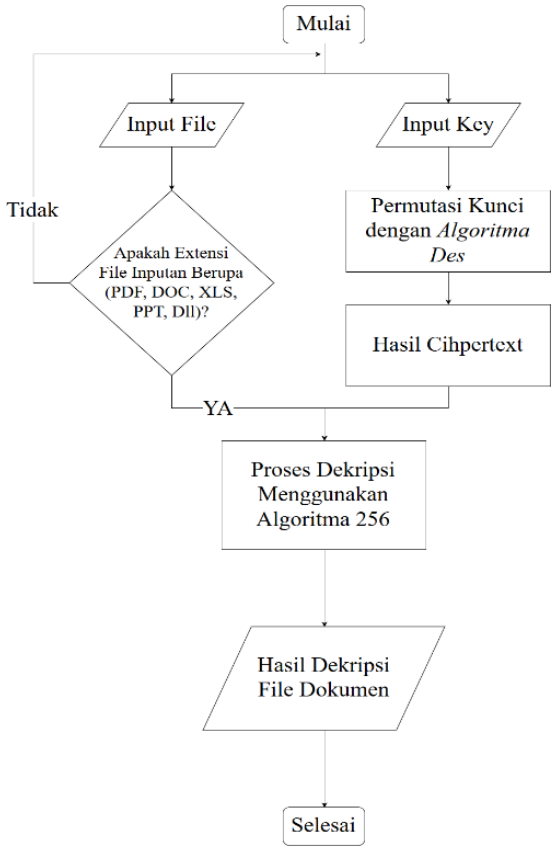
3.1 Proses Enkrip

Dapat dilihat pada Gambar 8 point a, proses yang terjadi pada saat enkripsi file adalah sebagai berikut :

a. Mulai proses dari enkripsi file.
b. Input file dan input kunci secara manual yang ingin diproses dengan metode kriptografi menggunakan algoritma AES dan DES.
c. Setelah file dan input kunci berhasil diinput, maka sistem akan mengecek apakah ekstensi file dan ukuran file tersebut cocok/benar?, Juka tidak maka akan kembali pada proses input file dan kunci. Jika ya, maka akan dilanjutkan pada proses selanjutnya.
d. Setelah melakukan pengecekan, maka kunci akan dienkripsi menggunakan permutasi kunci DES yang diputar sebanyak 16 kali, yang nanti mendapatkan bit biner dari hasil permutasi tersebut.
e. Hasil dari proses sebelumnya yaitu menggunakan ilmu kriptografi algoritma DES permutasi kunci untuk mendapatkan sebuah ciphertext.
f. Setelah dari proses sebelumnya yang menghasilkan ciphertext, program akan mengecek bahwa ciphertext tersebut masuk algoritma AES 256 yang nantinya digunakan sebagai kunci di algoritma AES tersebut.
g. Setelah dari proses yang menghasilkan kunci, program akan memukkan ciphertext hasil dari permutasi kunci kedalam algoritma AES yang digunakan untuk mengenkripsi file.
h. Setelah proses ke tujuh berhasil dilaksanakan, maka akan mengeluarkan output yangitu file hasil enkripsi file , yaitu file yang telah berhasil di enkripsi.
i. Akhir dari proses enkripsi file.



(a) Proses Encode atau Proses Enkripsi File



(b) Proses Decode atau Proses Dekripsi File

Gambar 8. Usulan metode

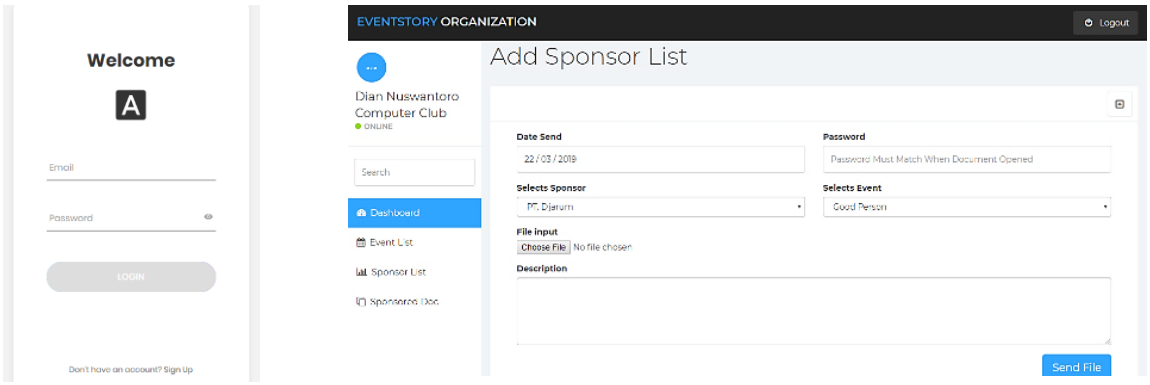
3.2 Proses Dekripsi atau Decode File

Pada proses dekripsi file, proses yang terjadi dapat dilihat pada Gambar 8 point b.

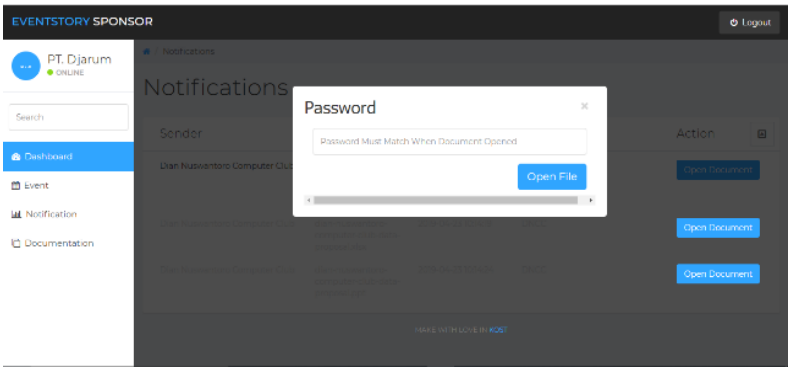
a. Mulai proses dari dekripsi file.
b. Input file dan input kunci yang sesuai seperti saat enkripsi file yang akan diproses dengan metode kriptografi menggunakan algoritma AES dan DES.
c. Setelah file dan input kunci berhasil diinput, maka sistem akan mengecek apakah ekstensi file dan ukuran file tersebut cocok/benar?, Jika tidak maka akan kembali pada proses input file dan kunci. Jika ya, maka akan dilanjutkan pada proses selanjutnya.
d. Setelah melakukan pengecekan, maka kunci akan dienkripsi menggunakan permutasi kunci DES yang diputar sebanyak 16 kali, yang nanti mendapatkan bit biner dari hasil permutasi tersebut.
e. Hasil dari proses sebelumnya yaitu menggunakan ilmu kriptografi algoritma DES permutasi kunci untuk mendapatkan sebuah ciphertext.
f. Setelah dari proses sebelumnya yang menghasilkan ciphertext, program akan mengecek bahwa ciphertext tersebut masuk algoritma AES 256 yang nantinya digunakan sebagai kunci di algoritma AES tersebut.
g. Setelah dari proses yang menghasilkan kunci, program akan memukkan ciphertext hasil dari permutasi kunci kedalam algoritma AES yang digunakan untuk mendekripsi file.
h. Setelah proses ke tujuh berhasil dilaksanakan, maka akan mengeluarkan output yaitu file hasil dekripsi file , yaitu file yang telah berhasil di dekripsi.

4. HASIL DAN PEMBAHASAN

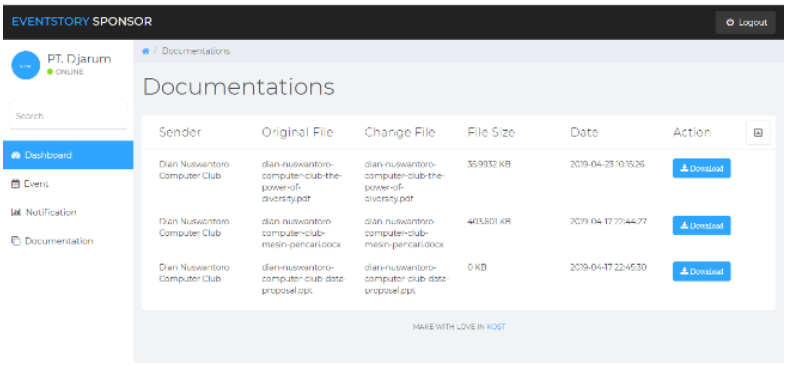
Arsitektur sistem yang digunakan untk membuat alikasi ini adalah 3 *tier*. Konsep arsitektur *three tier* (3 tier) atau yang biasa disebut dengan konsep client server *programming* merupakan konsep pemrograman yang terdiri dari 3 komponen *logic layer* yang saling berkaitan yaitu *presentation tier, logic tier dan database tier*. Arsitektur sistem tersebut. Tier pertama adalah presentation tier yang mana pada tier ini berisikan tampilan untuk interaksi antara pengguna dengan sistem. Pada tier kedia adalah logic tier, tier ini berfungsi sebagai pengeolahan data atau bertanggung jawab atas cara kerja aplikasi, pada *tier* ini data (file dokumen dan kunci) diolah menggunakan gabungan algoritma AES dan DES. Pada *tier* ketiga yaitu *database tier*, *tier* ini berfungsi sebagai penyimpanan data dari hasil yang diperoleh *logic tier*. Adapun yang disimpan oleh tier ini adalah data file dokumen, deskripsi data, dan tujuan pengiriman.



(a) (b)



(c)



(d)

Gambar 9. (a) Halaman Login, (b) Halaman Tambah Pengiriman Proposal Sponsor, (c) Tombol Open *Document* Sponsor, (d) Halaman *Documentation* File Sponsor

Menurut Gambar 9 (a), halaman login komunitas maupun sponsor yang digunakan untuk pengecekan apakan *authorization* tersebut benar atau salah yang mana setelah komunitas maupun sponsor memasukkan email dan password maka sistem akan memproses dan akan memindahkannya kehalaman *authorization*. Tampilan pada halaman ini sedikit berbeda dengan tampilan yang lainnya, dimana pada halaman ini tidak terdapat menu pada sistem. Sedangkan Gambar 9 (b) merupakan fungsi utama yang terdapat dalam sistem *Eventstory Crypto System* dimana pada halaman ini digunakan sebagai proses enkripsi file dokumen yang akan dikirimkan kepada sponsor, dimana komunitas perlu memasukkan kunci atau password, memilih pengajuan kepada sponsor, dan memilih event yang akan diajukan kepada sponsor, kemudian memilih file document dengan extensi .doc, .docx, .pdf, .ppt, .pptx, .xls, .xlsx yang nanti akan diproses lalu akan dikirimkan kepada sponsor yang dituju tampa diketahui oleh sponsor lainnya, terakhir memasukkan dekripsi dari pengiriman tersebut. Sedangkan Gambar 9 (c) merupakan halaman open document berfungsi untuk memasukkan password, yang mana password harus sama seperti pada saat file tersebut di enkripsi, lalu file tersebut akan diproses sedemikian rupa untuk menghasilkan plainfile atau file yang asli. Gambar diatas merupakan halaman dokumentasi atau kumpulan dari file – file yang dikirimkan oleh komunitas dan sudah berhasil dibuka oleh sponsor. Gambar 9 (d) memperlihatkan tombol download pada kolom action yang dapat digunakan oleh sponsor untuk mengunduh file asli dari file yang terlah dikirimkan komunitas.

Hasil eksperimen di evaluasi menggunakan *avalanche effect*, yang merupakan salah satu cara untuk menentukan baik atau tidaknya suatu algoritma kriptografi, dimana akan diketahui seberapa besar perubahan bit yang terjadi pada cipherteks akibat proses enkripsi. Semakin besar *avalanche effect* akan semakin baik algoritma kriptografi tersebut. *Avalanche Effect* dihitung dengan membandingkan perbedaan bit pada cipherteks 1 dan cipherteks 2. Eksperimen tersebut dilakukan pada plainfile yang sama dengan kunci yang berbeda. Dalam eksperimen ini akan melibatkan 3 buah file sebagai plainteks yang akan dienkripsi dalam beberapa kondisi. File-file tersebut adalah Proposal.docx, Text1.docx, dan Analisis.pdf yang berisikan karakter-karakter acak dengan ukuran file masing-masing 1 MB, 204 KB dan 159 KB.

Tabel 1. Daftar File Avalace Effect

Nama File	Ukuran File
Proposal.docx	1 MB
Text1.docx	204 KB
Analisis.pdf	159 KB

Untuk kunci akan digunakan 2 buah kunci, yaitu kunci 1 yang berupa kata ‘muhamadnabil123’, dan kunci 2 yang berupa kata ‘123nabilmuhamad’.

Tabel 2. Nilai Avalanve Effect

No	Nama File	Kunci	Jumlah Perbedaan Bit	Jumlah Keseluruhan Bit	Rata - Rata
1	Proposal.docx	Kunci 1	64	128	50%
		Kunci 2			
2	Text1.docx	Kunci 1	59	120	49.16%
		Kunci 2			
3	Analisis.pdf	Kunci 1	48	120	40%
		Kunci 2			
Rata – rata Keseluruhan					46.38%

Dari rata – rata 46.38% diatas dapat disimpulkan bahwa kombinasi antara algoritma DES dan AES merupakan algoritma yang aman dibuktikan dari pengujian *Avalance Effect* yang menghasilkan beberapa bit yang berbeda.

5. KESIMPULAN

Peneliti berhasil membuat sistem aplikasi bernama *Evenstory Crypto System* berbasis web untuk mengenkripsi file sehingga file dapat terlindungi dengan mengimplementasikan algoritma DES untuk verifikasi kunci dan AES-128 untuk enkripsi file. Sistem aplikasi berhasil mengenkripsi dan mendekripsi file dokumen teks, dengan ekstensi .docx, .doc, .pdf, .xls, .xlsx, .ppt, .pptx. File dokumen yang melaui proses enkripsi dan dekripsi akan mengalami perubahan besar file sebesar 0.05%, dikarenakan mengalami penambahan bit melalui proses enkripsi yang cukup panjang. Metode akan lebih lama dipecahkan oleh kriptanalis dengan alasan langkah untuk mengenkripsi file cukup banyak karena tidak hanya menggunakan satu metode tetapi lebih, selain itu kriptanalis tidak akan menyadari bahwa bit awal dari file yang terenkripsi merupakan hash kunci hasil fungsi DES. Dari pengujian *Avalanche Effect* kombinasi algoritma aes dan des dapat dikatakan aman dengan persentase 46.38%.

DAFTAR PUSTAKA

[1] M. Bellare and P. Rogaway, *Implementasi Affine Chipeh dan RC4 Pada Enkripsi Tunggal.*: Prosding SNATIF, 2015.

[2] Ratnadewi Ratnadewi, Adhie Roy Pramono, Yonatan Hutama, A. Saleh Ahmar, and M.I Setiawan, "Implementation Cryptography Data Encryption Standard (DES) and Triple Data Encryption Standard (3DES) Method in Communication System Based Near Field Communication (NFC)," *Journal of Physics: Conference Series*, vol. 954, no. 1, pp. 1-8, Januari 2018.

[3] Christy Atika Sari, Eko Hari Rachmawanto, and Edi Jaya Kusuma, "Good Performance Images Encryption Using Selective Bit T-des On Inverted Lsb Steganography," *Jurnal Ilmu Komputer dan Informasi*, vol. 12, no. 1, pp. 41-49, 2019.

[4] Albert Ginting and R. Rizal, Windasari, Ike Pertiwi Isnanto, "Implementasi Algoritma Kriptografi RSA untuk Enkripsi dan Dekripsi Email," *Jurnal Teknologi dan Sistem Komputer*, vol. 3, no. 2, pp. 253-258, April 2015.

[5] Kas Raygaputra Ilaga, Christy Atika Sari, and Eko Hari Rachmawanto, "A High Result for Image Security Using Crypto-Stegano Based on ECB Mode and LSB Encryption," *Journal of Applied Intelligent System*, vol. 3, no. 1, pp. 28-38, 2018.

[6] Andri. M. Yuli, *Implementasi Algoritma Kriptografi DES, RSA dan Algoritma Kompresi LZW pada Berkas Digital*. Sumatra Utara, 2009.

[7] Irjatul Wardah, "Kriptografi Pengiriman Image Pada Telephone Seluler Menggunakan Algoritma Des," *Jurnal Sistem Informasi (JSI)*, vol. 2, no. 3, pp. 371-387, Oktober 2011.

[8] Indra Syahputra, "Simulasi Kerahasiaan/Keamanan Informasi Dengan Menggunakan Algoritma Data Encryption Standart (DES)," Universitas Sumatra Utara, Sumatra Utara, Thesis 2009.

[9] Mukund R. Joshi and Renuka Avinash Karkade, "Network Security with Cryptography," *International Journal of Computer Science and Mobile Computing*, vol. 4, no. 1, pp. 201-204, Januari 2015.

[10] Sari. Christy Attika, Eko Hari Rachamwanto, and Edi Jaya Kusuma, "Good Performance Images Encryption Using Selective Bit T-des On Inverted Lsb Steganography," *Jurnal Ilmu Komputer dan Informasi (JIKI)*, vol. 12, no. 1, pp. 41-49, 2019.

[11] Eko Hari Rachmawanto and Christy Atika Sari, "Keamanan File Menggunakan Teknik Kriptografi Shift Cipher," *Techno.COM*, vol. 14, no. 4, pp. 329-335, 2015.

[12] M.Pitchaiah, Philemon Daniel, and Praveen Praveen, "Implementation of Advanced Encryption Standard Algorithm," *International Journal of Scientific & Engineering Research*, vol. 3, no. 3, pp. 1-6, Maret 2012.

[13] Eko Hari Rachmawanto, Rofi' Syaiful Amin, D.R.I.M. Setiadi, and Christy Atika Sari, "A performance analysis StegoCrypt algorithm based on LSB-AES 128 bit in various image size," in *International Seminar on Application for Technology of Information and Communication*, Semarang, 2017, pp. 16-21.

[14] Fahmi Anwar, Eko Hari Rachmawanto, Christy Atika Sari, and D.R.I.M Setiadi, "StegoCrypt Scheme using LSB-AES Base64," in *International Conference on Information and Communications Technology (ICOIACT)*, Yogyakarta, 2019, pp. 85-90.