IMPLEMENTASI KRIPTOGRAFI KUNCI PRIVAT UNTUK KEAMANAN LAMPIRAN SURAT ELEKTRONIK

Veronica Lusiana, Budi Hartono

Program studi Teknik Informatika, Fakultas Teknologi Informasi,
Universitas Stikubank Semarang, Jl. Tri Lomba Juang No.1 Mugas, Semarang
verolusiana@yahoo.com, pakbudi@yahoo.com

Abstrak

Penyadapan informasi merupakan hal yang merugikan bagi pengguna layanan surat elektronik (*e-mail*), sehingga aspek keamanan saat pertukaran informasi sangat penting untuk dipertimbangkan. Secara umum upaya peningkatan keamanan dapat dilakukan terhadap isi pesan surat elektronik (*e-mail*) dan berkas lampiran (*file attachment*) yang menyertai surat elektronik. Tujuan penelitian ini adalah untuk meningkatkan keamanan berkas lampiran yang dikirimkan bersama surat elektronik. Algoritma kriptografi yang akan digunakan adalah algoritma kunci privat AES (*Advanced Encryption Standard*). Algoritma AES digunakan untuk menyandikan berkas lampiran surat elektronik. Berkas lampiran yang digunakan adalah dokumen jenis: txt, doc, xls, ppt, pdf, odt, ods, odp, bmp, jpg dan exe. Analisa dan pembahasan adalah dari ukuran berkas baru yang dihasilkan dan waktu yang dibutuhkan untuk proses enkripsi dan dekripsi terhadap beberapa jenis dokumen tersebut dengan ukuran berkas yang bervariasi. Agar hasil penelitian ini dapat bermanfaat secara langsung maka akan diimplementasikan menjadi prototipe perangkat lunak komputer. Melalui perangkat lunak ini diharapkan dapat meningkatkan keamanan berkas lampiran surat elektronik sehingga hanya dapat dibaca atau dibuka oleh yang berhak saja.

Kata kunci: surat elektronik, kunci privat, Algoritma AES

1. PENDAHULUAN

Penyadapan informasi merupakan hal yang merugikan bagi pengguna Internet khususnya yang memanfaatkan layanan surat elektronik (*e-mail*), sehingga aspek keamanan saat pertukaran informasi sangat penting untuk dipertimbangkan. Penelitian ini fokus pada cara untuk meningkatkan keamanan pertukaran informasi melalui surat elektronik. Secara umum upaya ini dapat dilakukan dengan menjaga kerahasiaan informasi. Informasi disini adalah dalam bentuk lampiran berkas digital (*file attachment*).

Telah banyak dikembangkan algoritma kriptografi yang dapat dipakai untuk meningkatkan keamanan data. Kriptografi menyediakan layanan untuk meningkatkan keamanan data seperti: kerahasiaan (*confidentiality* atau *secrecy*), otentikasi (*authentication*), keaslian pesan (*data integrity*), dan anti penyangkalan (*non-repudiation*) [7]. Kerahasiaan adalah layanan yang ditujukan untuk menjaga agar informasi tidak dapat diketahui oleh pihak yang tidak berhak. Otentikasi adalah layanan untuk mengidentifikasi kebenaran pihak-pihak yang berkomunikasi (*user authentication*) beserta kebenaran sumber pesan (*data origin authentication*). Keaslian pesan adalah layanan untuk memastikan data masih asli atau utuh, belum pernah dimanipulasi selama proses pengiriman. Sedangkan anti penyangkalan merupakan layanan untuk mencegah pelaku komunikasi yaitu pihak pengirim maupun penerima melakukan penyangkalan [3].

Aritkel ini membahas tentang hal-hal sebagai berikut:

1. Kajian penerapan algoritma kunci privat AES untuk menyandikan berkas lampiran surat elektronik.
2. Analisa waktu yang dibutuhkan untuk proses enkripsi dan dekripsi terhadap beberapa jenis berkas dokumen dengan ukuran berkas yang bervariasi.

Batasan masalah pada penelitian ini adalah sebagai berikut:

1. Algoritma kriptografi yang digunakan adalah AES (*Advanced Encryption Standard*, AES-128, AES-192, dan AES-256).
2. Berkas lampiran yang disandikan menggunakan algoritma AES adalah dokumen jenis: txt, doc, xls, ppt, pdf, odt, ods, odp, bmp, jpg dan exe). Ukuran maksimum berkas adalah 500 KB.
3. Melakukan pengujian hasil proses enkripsi dan dekripsi untuk memastikan berkas atau data dapat disandikan dan dapat dikembalikan ke dalam bentuk aslinya.
4. Mencatat waktu yang diperlukan dan ukuran berkas yang dihasilkan dari proses enkripsi dan dekripsi.

Tujuan penelitian ini adalah meningkatkan keamanan berkas lampiran yang dikirimkan bersama surat elektronik menggunakan algoritma kriptografi

kunci privat AES (*Advanced Encryption Standard*). Algoritma AES digunakan untuk menyandikan berkas lampiran yang akan dikirimkan bersama surat elektronik. Agar hasil penelitian ini dapat bermanfaat secara langsung maka akan diimplementasikan menjadi prototipe perangkat lunak komputer. Perangkat lunak ini diharapkan dapat digunakan secara mudah oleh pengguna tanpa pelatihan khusus.

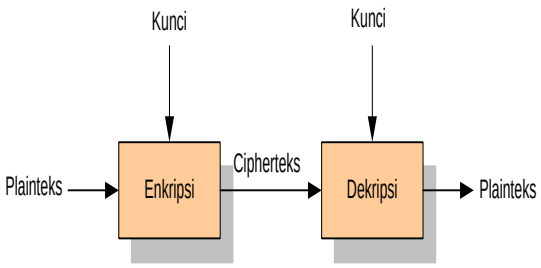
2. TINJAUAN PUSTAKA

Kriptografi

Kriptografi adalah ilmu untuk menjaga keamanan data atau informasi yang meliputi kerahasiaan, integritas, serta otentikasi [7]. Kriptografi bertujuan mengamankan data dari orang yang tidak berhak untuk mengetahui isi data tersebut. Beberapa istilah yang berhubungan dengan kriptografi [3]:

a. *Plainteks (Plaintext* atau *cleartext*) adalah data asli atau informasi bersifat terbuka yang isinya dapat dibaca dan dipahami secara langsung.
b. *Cipherteks* (*Ciphertext*) adalah data sandi hasil proses enkripsi.
c. *Cipher* adalah algoritma untuk mengubah *plainteks* menjadi *cipherteks*.
d. *Kunci* atau *key* adalah data nilai yang sangat spesifik, kunci diketahui oleh pengirim dan penerima yang berhak. Digunakan bersama dengan algoritma kriptografi untuk melakukan proses enkripsi dan dekripsi.
e. *Enkripsi* atau *encryption* adalah proses untuk mengubah plainteks menjadi cipherteks, yaitu proses menyandikan atau menyembunyikan *plainteks*.
f. *Dekripsi* atau *decryption* adalah proses untuk mengembalikan *cipherteks* menjadi *plainteks*, proses kebalikan dari enkripsi.

Algoritma kriptografi adalah fungsi matematis yang digunakan untuk proses enkripsi yaitu proses untuk menyandikan pesan atau informasi (menjadi bentuk *ciphertext*), dan proses dekripsi yaitu proses untuk membuka suatu pesan tersandi menjadi pesan semula (*plaintext*) [4]. Pada Gambar 1 dapat dilihat proses kriptografi secara umum yaitu enkripsi dan dekripsi untuk pesan terbuka (M) dan pesan rahasia (C), dimana M dan C bisa berukuran sama atau berbeda.



Gambar 1 Skema proses enkripsi dan dekripsi

Kriptografi algoritma kunci simetris (kunci privat)

Pembentukan kunci pada algoritma kunci simetris dan asimetris disusun dari sejumlah karakter dengan panjang tertentu. Tingkat keamanan dari algoritma yang menggunakan kunci adalah berdasarkan kerahasiaan kuncinya bukan berdasarkan algoritmanya, karena algoritma bersifat terbuka. Semakin panjang dan beragam kombinasi karakter yang digunakan sebagai kunci maka kerahasiaan kunci semakin aman.

Kriptografi algoritma simetris dikenal dengan istilah algoritma kunci privat (*private key algorithm*), dapat disebut juga dengan istilah *one-key algorithm*, *secret-key algorithm*, atau *single-key algorithm*. Algoritma ini memiliki kunci enkripsi dan dekripsi yang sama, yaitu $K_1=K_2=K$. Kunci K harus dirahasiakan dari pihak yang tidak berkepentingan sehingga tingkat keamanannya ditentukan oleh kerahasiaan kunci K. Rumus algoritma ini dapat dituliskan seperti pada Rumus 2.1 dan Rumus 2.2 [6].

$$e_K (M) = C \quad \text{..... (2.1)}$$

$$d_K (C) = M \quad \text{..... (2.2)}$$

Beberapa contoh algoritma simetris yaitu: DES (*Data Encryption Standard*), S-DES (*Simplified Data Encryption Standard*), AES (*Advanced Encryption Standard*), dan IDEA (*International Data Encryption Algorithm*).

Algoritma AES

Algoritma AES mengunakan kunci yang sama untuk proses enkripsi dan dekripsi. Algoritma AES memiliki tiga versi yaitu: AES-128, AES-192 dan AES-256, masing-masing menggunakan kunci internal (*round key*) yang berbeda untuk setiap proses putaran [1][5]. Proses putaran enkripsi AES-128 dikerjakan 10 kali (a=10), yaitu sebagai berikut:

1. Addroundkey
2. Putaran sebanyak a-1 kali, dimana setiap putaran dilakukan proses: *SubBytes*, *ShiftRows*, *MixColumns*, dan *AddRoundKey*.
3. Final round, adalah proses untuk putaran terakhir yang meliputi: *SubBytes*, *ShiftRows*, dan *AddRoundKey*.

Pada proses dekripsi AES-128, proses putaran dikerjakan 10 kali (a=10), yaitu:

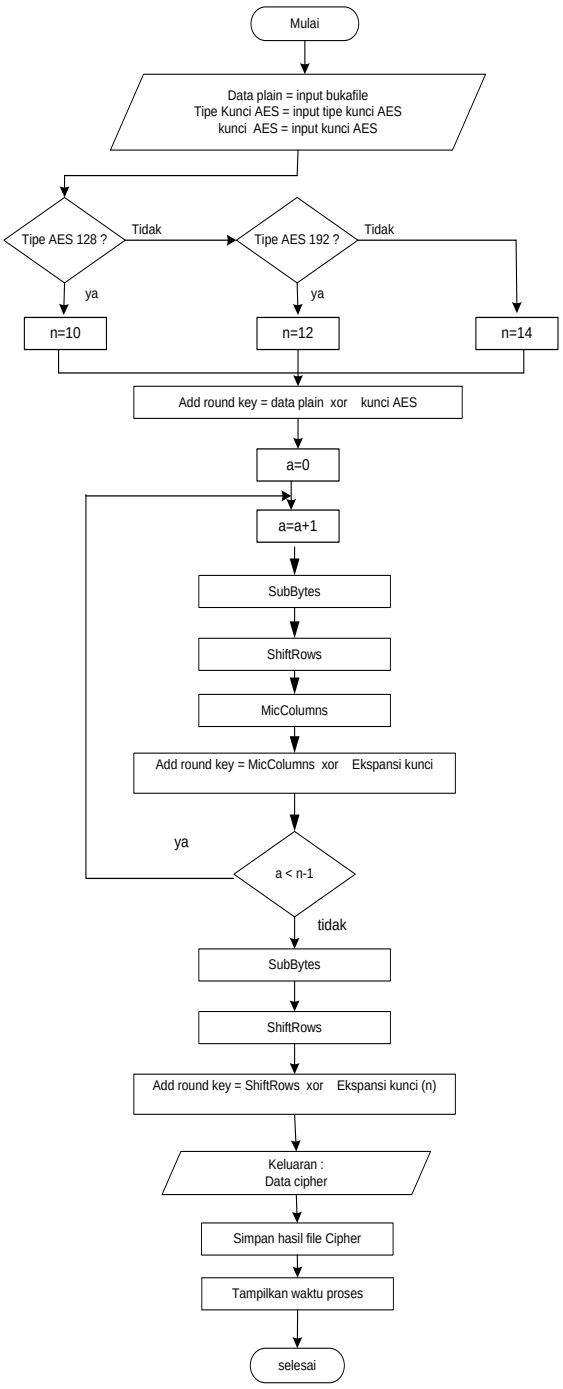
1. Addroundkey
2. Putaran sebanyak a-1 kali, dimana setiap putaran dilakukan proses: *InverseShiftRows*, *InverseSubBytes*, *AddRoundKey*, dan *InverseMixColumns*.
3. Final round, adalah proses untuk putaran terakhir yang meliputi: *InverseShiftRows*, *InverseSubBytes*, dan *AddRoundKey*.

Pada enkripsi dan dekripsi AES-192 proses putaran dikerjakan 12 kali (a=12), untuk AES-256 proses putaran dikerjakan 14 kali (a=14). Pada Tabel 1 menjelaskan perbedaan versi AES tersebut. Pada

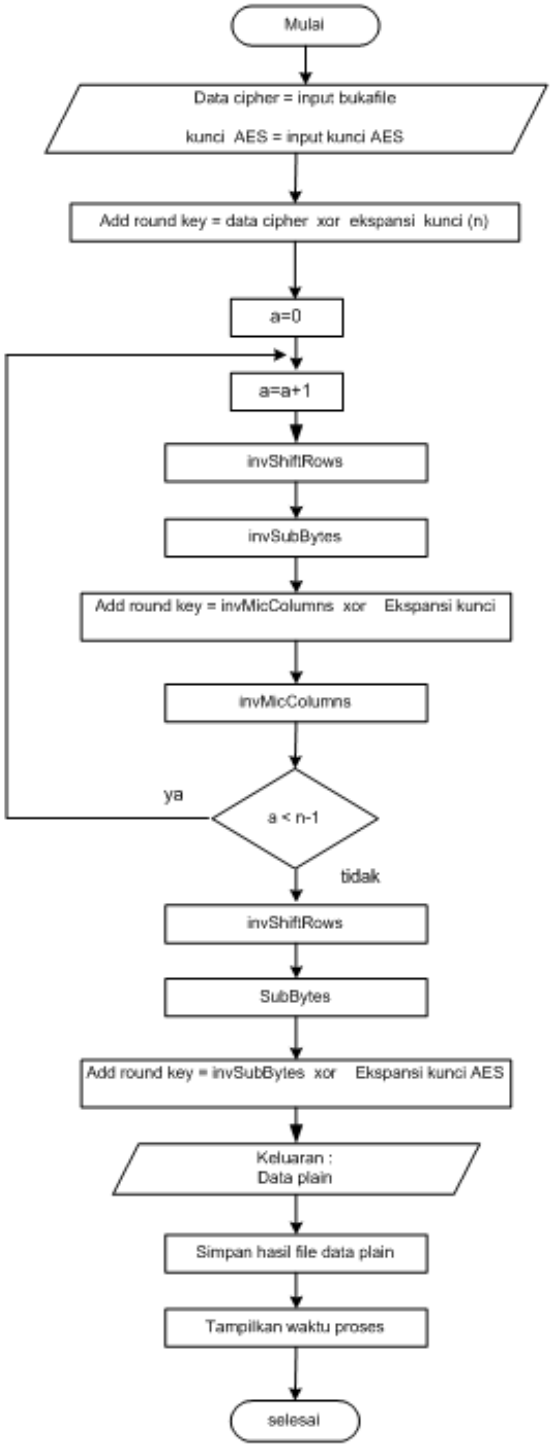
Gambar 3 dan Gambar 3 dapat dilihat diagram alir proses enkripsi dan dekripsi algoritma AES.

Tabel 1. Tiga buah versi AES

	Panjang kunci (Nk words)	Ukuran Blok (Nb words)	Jumlah Putaran (Nr)
AES-128	4	4	10
AES -192	6	4	12
AES-256	8	4	14



Gambar 2 Diagram alir enkripsi algoritma AES



Gambar 3. Diagram alir dekripsi algoritma AES

3. METODOLOGI PENELITIAN

Langkah-langkah pada penelitian ini adalah sebagai berikut:

1. Analisa masalah dan studi literatur.
   Masalah yang menjadi obyek pengamatan adalah meningkatkan keamanan pengiriman berkas lampiran melalui surat elektronik. Studi literatur dilakukan untuk mencari referensi yang mendukung penelitian ini.

2. Menentukan metode yang digunakan untuk mengatasi masalah. Metode yang dipilih adalah kriptografi kunci privat (algoritma AES).
3. Merancang model prototipe perangkat lunak.
4. Uji coba proses enkripsi dan dekripsi menggunakan berbagai jenis berkas dokumen dan mengamati tingkat keberhasilannya.
5. Analisa hasil terhadap kebutuhan waktu proses dan ukuran berkas yang dihasilkan.

Alat dan Bahan Penelitian

Alat penelitian meliputi perangkat keras komputer dengan spesifikasi processor Intel Core I3-2350M, memory 2GB, hard disk 500GB, Sistem Operasi Windows 7 32 bit dan perangkat lunak Matlab. Bahan penelitian meliputi beberapa jenis berkas dengan ukuran yang bervariasi, seperti dapat dilihat pada Tabel 2.

Tabel 2. Berkas yang digunakan untuk penelitian

No	Nama berkas	Jenis berkas	Ukuran berkas (byte)
1	coba_1	dokumen_txt	96
2	coba_2	dokumen_txt	183
3	coba_3	dokumen_doc	230.400
4	coba_4	dokumen_doc	214.528
5	coba_5	tabel_xls	28.160
6	coba_6	tabel_xls	76.800
7	coba_7	presentasi_ppt	476.160
8	coba_8	presentasi_ppt	526.848
9	coba_9	dokumen_pdf	23.907
10	coba_10	dokumen_pdf	35.572
11	coba_11	dokumen_odt	59.593
12	coba_12	dokumen_odt	14.222
13	coba_13	tabel_ods	15.225
14	coba_14	tabel_ods	17.984
15	coba_15	presentasi_odp	207.460
16	coba_16	presentasi_odp	239.614
17	coba_18	citra_bmp	15.478
18	coba_20	citra_jpg	32.413
19	coba_22	animasi_exe	518.161

4. HASIL DAN PEMBAHASAN

Menyusun berkas hasil enkripsi

Berkas hasil enkripsi (*.cip) disusun dari dua komponen yaitu: *data header* dan *data cipher*. *Informasi header* terdiri dari 8 karakter identitas yaitu "*CARAv1 x"* dengan karakter *x* mencatat jenis AES (AES-128, AES-192, atau AES-256), dan ditambah dengan kunci AES yang telah diacak (sebanyak 16, 24, atau 32 karakter) sesuai dengan jenis AES yang digunakan [2]. Informasi header ini sebagai pengenal berkas hasil enkripsi dan digunakan untuk mendeteksi benar atau salah kunci yang digunakan pada awal proses dekripsi.

Sesuai dengan ketentuan algoritma AES, data yang dienkripsi adalah dalam kelipatan 16 byte atau 16 karakter agar algoritma ini dapat bekerja dengan baik. Pada penelitian ini sumber data yang dienkripsi berasal dari berkas dengan berbagai macam ukuran seperti dapat dilihat pada Tabel 2. Untuk memenuhi ketentuan tersebut maka ukuran data yang akan dienkripsi perlu disesuaikan dengan menambah karakter secukupnya agar menjadi kelipatan 16 byte. Pada penelitian ini, data ditambah dengan karakter sisipan berupa '~' sebanyak 1 sampai dengan 15 buah sesuai kebutuhan yang diikuti dengan sebuah karakter yang mencatat jumlah penambahan karakter '~'. Karakter sisipan diletakkan di akhir data.

Berikut ini contoh perhitungan ukuran berkas hasil proses enkripsi dengan tiga jenis AES [2]. Contoh untuk berkas dokumen txt coba_1 yang berukuran 96 byte, maka ukuran berkas enkripsi adalah sebesar:

a. Pada AES-128: 8 (identitas) + 16 (kunci yang diacak) + 96 (plain) + 15 ('~') + 1 = 136
b. AES-192: 8 (identitas) + 24 (kunci yang diacak) + 96 (plain) + 15 ('~') + 1 = 144
c. AES-256: 8 (identitas) + 32 (kunci yang diacak) + 96 (plain) + 15('~') + 1 = 152

Catatan: 96 (plain) + 15 ('~') + 1 = 112, 112 mod 16 = 0 (kelipatan 16).

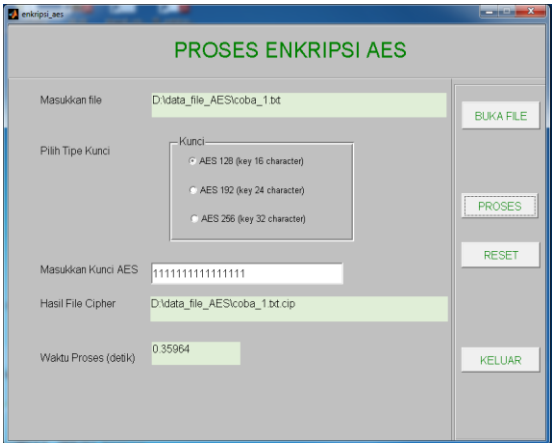
Hasil enkripsi dan dekripsi berkas lampiran surat elektronik

Proses enkripsi dan dekripsi berkas lampiran surat elektronik dikerjakan menggunakan algoritma AES-128, AES-192 dan AES-256. Proses analisa meliputi:

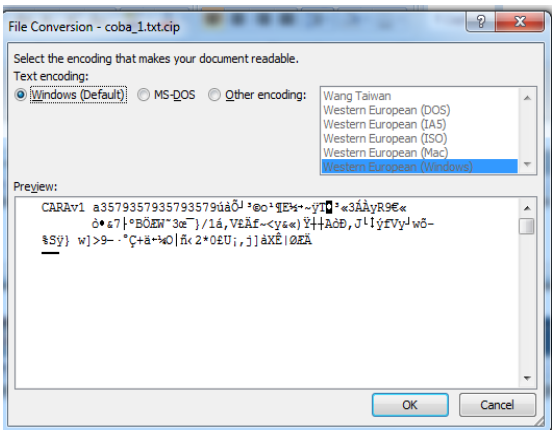
a. Waktu yang diperlukan untuk proses enkripsi dan dekripsi.
b. Ukuran berkas yang dihasilkan dari proses enkripsi dan proses dekripsi. Proses enkripsi menghasilkan berkas baru dalam bentuk terenkripsi (berkas *.cip).

Tampilan menu proses enkripsi dan dekripsi untuk berkas lampiran dapat dilihat pada gambar 4 dan gambar 6. Pada gambar 5 adalah tampilan hasil proses enkripsi dan gambar 7 adalah tampilan hasil proses dekripsi.

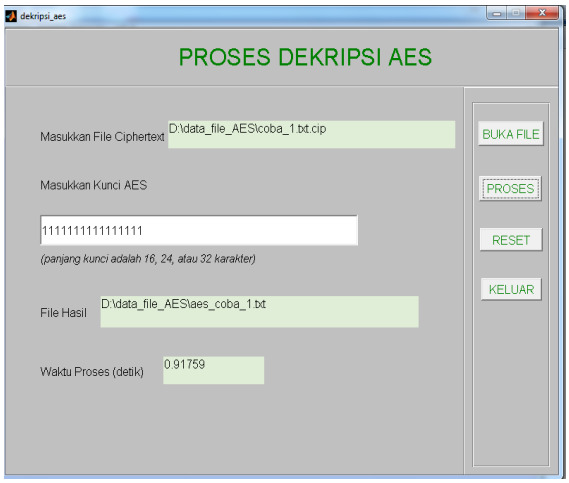
Pada Tabel 2 mencatat ukuran berkas hasil enkripsi dan dekripsi AES-128 beserta waktu proses yang diperlukan, menggunakan tiga contoh kunci yaitu *111111111111111*, *1234567890123456*, atau *veronica_ lusiana*. Berkas coba_1.txt (96 byte) dalam bentuk cipher membutuhkan ruang penyimpan 136 byte, dengan waktu proses enkripsi dan dekripsi tercepat dibandingkan berkas lain yaitu 0,35 dan 0,91 detik. Sedangkan berkas coba_8.ppt (526.848 byte) dalam bentuk cipher berukuran 526.877 byte, dengan waktu proses enkripsi dan dekripsi terlama yaitu 2.551 dan 7.653 detik.



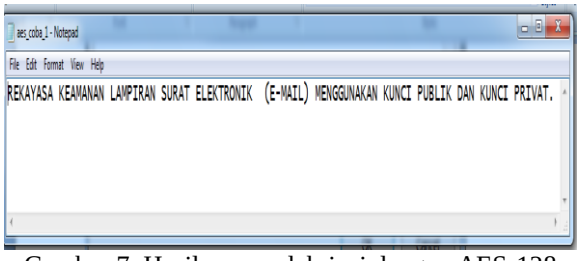
Gambar 4. Menu proses enkripsi dengan AES-128



Gambar 5. Hasil proses enkripsi AES-128



Gambar 6. Menu proses dekripsi dengan AES-128



Gambar 7. Hasil proses dekripsi dengan AES-128

Pada Tabel 3 mencatat ukuran berkas hasil enkripsi dan dekripsi AES-192 beserta waktu proses yang diperlukan, menggunakan dua contoh kunci yaitu *111111111111111111111*, atau *000000000000000000000*. Berkas coba_1.txt dalam bentuk cipher membutuhkan ruang penyimpan 144 byte, dengan waktu proses enkripsi dan dekripsi tercepat dibandingkan berkas lain yaitu 0,4 dan 1 detik. Sedangkan berkas coba_8.ppt dalam bentuk cipher berukuran 526.885 byte, dengan waktu proses enkripsi dan dekripsi terlama yaitu 2.608 dan 7.824 detik.

Pada Tabel 4 mencatat ukuran berkas hasil enkripsi dan dekripsi AES-256 beserta waktu proses yang diperlukan, menggunakan sebuah contoh kunci yaitu *veronica_lusiana_budi_hartono123*. Berkas coba_1.txt dalam bentuk cipher membutuhkan ruang penyimpan 152 byte, dengan waktu proses enkripsi dan dekripsi tercepat dibandingkan berkas lain yaitu 0,83 detik dan 1 detik. Sedangkan berkas coba_8.ppt dalam bentuk cipher berukuran 526.893 byte, dengan waktu proses enkripsi dan dekripsi terlama yaitu 2.675 dan 8.025 detik.

4.3 Kompleksitas kebutuhan ruang penyimpan berkas dan waktu proses

Pada Tabel 3 , Tabel 4, dan Tabel 5 dapat diamati bahwa ukuran berkas hasil enkripsi AES dan kebutuhan waktu prosesnya berbanding lurus dengan ukuran berkas asli (*plain*). Ukuran berkas hasil enkripsi dan kebutuhan waktu proses dipengaruhi oleh ukuran berkas asli, namun tidak dipengaruhi oleh jenis berkas. Semakin besar ukuran berkas asli maka semakin besar pula ukuran berkas hasil enkripsi dan kebutuhan waktu prosesnya. Hal tersebut berlaku bagi ketiga jenis AES.

Pada proses dekripsi, informasi header dan karakter sisipan dihilangkan. Pada Tabel 3 , Tabel 4, dan Tabel 5 dapat dilihat ukuran berkas hasil proses dekripsi adalah sama seperti ukuran berkas aslinya. Proses dekripsi algoritma AES memerlukan komputasi yang lebih kompleks jika dibandingkan dengan proses enkripsi, sehingga menyebabkan waktu yang dibutuhkan untuk proses dekripsi menjadi lebih lama.

Pada Tabel 6 mencatat ukuran berkas hasil enkripsi menggunakan tiga jenis AES beserta rasionya. Rasio selisih ukuran berkas hasil enkripsi diperoleh dari ((berkas hasil enkripsi - berkas plain) / berkas plain) * 100%. Dari tabel tersebut, baris ke tiga dan seterusnya dapat dilihat ukuran berkas hasil enkripsi tidak berbeda jauh dengan berkas aslinya. Prosentase rasionya cukup kecil yaitu kurang dari satu persen, hal ini terjadi karena jumlah penambahan data header dan karakter sisipan pada berkas hasil enkripsi adalah relatif kecil, untuk berapapun ukuran berkas aslinya. Pada dua baris pertama dari tabel ini prosentase rasio cukup besar dikarenakan ukuran berkas asli yang kecil. Secara umum semakin besar ukuran berkas asli maka rasionya menjadi semakin

kecil. Dari hasil analisa di atas maka algoritma AES cocok untuk mengenkripsi data dengan ukuran yang besar karena data hasil enkripsi mendekati ukuran data aslinya.

Dengan spesifikasi komputer yang digunakan pada penelitian ini maka proses enkripsi mulai terasa lama untuk berkas asli yang berukuran 35.572 byte atau lebih, yaitu memerlukan waktu lebih dari satu menit. Apabila diamati dari Tabel 7 maka prosentase rasio waktu dekripsi dibagi waktu enkripsi akan berada pada kisaran 120% sampai dengan 450%. Rasio diperolah dari (waktu dekripsi / waktu enkripsi) * 100%. Rata-rata kebutuhan waktu proses dekripsi adalah antara tiga sampai dengan empat setengah kali lebih lama dibandingkan dengan proses enkripsi. Kebutuhan waktu proses disini adalah relatif yaitu dipengaruhi oleh kemampuan komputasi komputer dan seberapa banyak aplikasi yang sedang dijalankan secara bersamaan pada saat proses enkripsi dan dekripsi berlangsung.

5. KESIMPULAN DAN SARAN

Kesimpulan

Ukuran berkas hasil enkripsi (*.cip) adalah berbanding lurus dengan ukuran berkas aslinya dan panjang kunci AES yang digunakan, dengan penambahan ukuran berkas *.cip yang relatif kecil. Dari hasil percobaan diperoleh rata-rata penambahan ukuran berkas *.cip kurang dari 1%. Variasi jenis berkas tidak berpengaruh langsung terhadap ukuran berkas *.cip yang dihasilkan.

Komputasi proses dekripsi berlangsung lebih banyak jika dibandingkan dengan proses enkripsi, sehingga kebutuhan waktu proses dekripsi menjadi lebih lama dibandingkan dengan proses enkripsi. Dari hasil percobaan diperoleh rata-rata waktu proses dekripsi adalah 3 sampai dengan 4,5 kali lebih lama dari proses enkripsi. Variasi jenis berkas tidak berpengaruh langsung terhadap waktu yang diperlukan untuk kedua proses ini.

Saran

Untuk berkas lampiran yang berukuran relatif besar sebelum proses enkripsi akan lebih baik apabila dikompres terlebih dulu, hal ini berguna untuk mempercepat proses enkripsi dan dekripsi menggunakan algoritma AES. Agar program aplikasi ini dapat digunakan oleh masyarakat luas secara bebas, maka sebaiknya program aplikasi memiliki lisensi *free software* serta bersifat *stand alone.*

Tabel 3. Hasil proses enkripsi dan dekripsi AES-128

No	Nama berkas	Jenis berkas	Ukuran berkas (byte)			Waktu proses (detik)	
			plain	enkripsi	dekripsi	enkripsi	dekripsi
1	coba_1	dokumen txt	96	136	96	0,35	0,91
2	coba_2	dokumen txt	183	216	183	0,41	1
3	coba_12	dokumen odt	14.222	14.248	14.222	24	102
4	coba_13	tabel ods	15.225	15.250	15.225	26	109
5	coba_18	citra bmp	15.478	15.502	15.478	26	112
6	coba_14	tabel ods	17.984	18.011	17.984	30	129
7	coba_9	dokumen pdf	23.907	23.942	23.907	40	173
8	coba_5	tabel xls	28.160	28.200	28.160	47	203
9	coba_20	citra jpg	32.413	32.440	32.413	55	236
10	coba_10	dokumen pdf	35.572	35.600	35.572	64	274
11	coba_11	dokumen_odt	59.593	59.625	59.593	104	447
12	coba_6	tabel xls	76.800	76.840	76.072	137	596
13	coba_15	presentasi odp	207.460	207.499	207.460	520	2.022
14	coba_4	dokumen doc	214.528	214.568	214.528	559	2.132
15	coba_3	dokumen doc	230.400	230.440	230.400	613	2.333
16	coba_16	presentasi odp	239.614	239.640	239.614	657	2.454
17	coba_7	presentasi ppt	476.160	476.200	476.160	2.187	6.636
18	coba_22	animasi exe	518.161	518.190	518.161	2.321	6.963
19	coba_8	presentasi ppt	526.848	526.877	526.848	2.551	7.653

Tabel 4. Hasil proses enkripsi dan dekripsi AES-192

No	Nama berkas	Jenis berkas	Ukuran berkas (byte)			Waktu proses (detik)	
			plain	enkripsi	dekripsi	enkripsi	dekripsi
1	coba_1	dokumen txt	96	144	96	0,4	1
2	coba_2	dokumen txt	183	224	183	0,8	2
3	coba_12	dokumen odt	14.222	14.256	14.222	28	121
4	coba_13	tabel ods	15.225	15.258	15.225	30	130
5	coba_18	citra bmp	15.478	15.510	15.478	30	132
6	coba_14	tabel ods	17.984	18.019	17.984	35	154
7	coba_9	dokumen pdf	23.907	23.950	23.907	47	204
8	coba_5	tabel xls	28.160	28.208	28.160	55	241
9	coba_20	citra jpg	32.413	32.448	32.413	63	278
10	coba_10	dokumen pdf	35.572	35.608	35.572	74	324
11	coba_11	dokumen_odt	59.593	59.633	59.593	119	528
12	coba_6	tabel xls	76.800	76.848	76.072	159	696
13	coba_15	presentasi odp	207.460	207.507	207.460	576	2.292
14	coba_4	dokumen doc	214.528	214.576	214.528	604	2.405
15	coba_3	dokumen doc	230.400	230.448	230.400	670	2.638
16	coba_16	presentasi odp	239.614	239.648	239.614	728	2.773
17	coba_7	presentasi ppt	476.160	476.208	476.160	2.244	6.732
18	coba_22	animasi exe	518.161	518.198	518.161	2.378	7.134
19	coba_8	presentasi ppt	526.848	526.885	526.848	2.608	7.824

Tabel 5. Hasil proses enkripsi dan dekripsi AES-256

No	Nama berkas	Jenis berkas	Ukuran berkas (byte)			Waktu proses (detik)	
			plain	enkripsi	dekripsi	enkripsi	dekripsi
1	coba_1	dokumen txt	96	152	96	0,83	1
2	coba_2	dokumen txt	183	232	183	0,9	3
3	coba_12	dokumen odt	14.222	14.264	14.222	31	140
4	coba_13	tabel ods	15.225	15.266	15.225	34	150
5	coba_18	citra bmp	15.478	15.518	15.478	34	152
6	coba_14	tabel ods	17.984	18.027	17.984	40	177
7	coba_9	dokumen pdf	23.907	23.958	23.907	53	235
8	coba_5	tabel xls	28.160	28.216	28.160	62	278
9	coba_20	citra jpg	32.413	32.456	32.413	72	320
10	coba_10	dokumen pdf	35.572	35.616	35.572	83	373
11	coba_11	dokumen_odt	59.593	59.641	59.593	133	603
12	coba_6	tabel xls	76.800	76.856	76.072	177	797
13	coba_15	presentasi odp	207.460	207.515	207.460	618	2.565
14	coba_4	dokumen doc	214.528	214.584	214.528	649	2.681
15	coba_3	dokumen doc	230.400	230.456	230.400	727	2.924
16	coba_16	presentasi odp	239.614	239.656	239.614	766	3.067
17	coba_7	presentasi ppt	476.160	476.216	476.160	2.311	6.933
18	coba_22	animasi exe	518.161	518.206	518.161	2.445	7.335
19	coba_8	presentasi ppt	526.848	526.893	526.848	2.675	8.025

Tabel 6. Hasil proses enkripsi dan rasio selisih ukuran berkas

No	Nama berkas	Ukuran berkas (byte)				Rasio selisih ukuran berkas (%)		
		Plain	terenkripsi					
			AES-128	AES-192	AES-256	AES-128	AES-192	AES-256
1	coba_1	96	136	144	152	41,666	50	58,33
2	coba_2	183	216	224	232	18,032	22,404	26,77
3	coba_12	14.222	14.248	14.256	14.264	0,1828	0,239	0,2953
4	coba_13	15.225	15.250	15.258	15.266	0,1642	0,2167	0,269
5	coba_18	15.478	15.502	15.510	15.518	0,155	0,2067	0,258
6	coba_14	17.984	18.011	18.019	18.027	0,1501	0,1946	0,239
7	coba_9	23.907	23.942	23.950	23.958	0,1426	0,1798	0,213
8	coba_5	28.160	28.200	28.208	28.216	0,142	0,1704	0,198
9	coba_20	32.413	32.440	32.448	32.456	0,0832	0,1079	0,132
10	coba_10	35.572	35.600	35.608	35.616	0,0787	0,1012	0,123
11	coba_11	59.593	59.625	59.633	59.641	0,0536	0,0671	0,08
12	coba_6	76.800	76.840	76.848	76.856	0,052	0,0625	0,07
13	coba_15	207.460	207.499	207.507	207.515	0,01867	0,0226	0,0265
14	coba_4	214.528	214.568	214.576	214.584	0,01864	0,0223	0,0261
15	coba_3	230.400	230.440	230.448	230.456	0,0173	0,0208	0,024
16	coba_16	239.614	239.640	239.648	239.656	0,0108	0,0141	0,0175
17	coba_7	476.160	476.200	476.208	476.216	0,0084	0,01	0,0117
18	coba_22	518.161	518.190	518.198	518.206	0,005596	0,0071	0,0086
19	coba_8	526.848	526.877	526.885	526.893	0,005504	0,007	0,0085

Tabel 7. Hasil proses enkripsi – dekripsi dan rasio waktu proses

No	Nama berkas	Ukuran berkas plain (byte)	waktu proses (detik) (E=enkripsi, D=dekripsi)						rasio waktu proses (%) (dekripsi:enkripsi)*100%		
			E AES-128	D AES-128	E AES-192	D AES-192	E AES-256	D AES-256	AES-128	AES-192	AES-256
1	coba_1	96	0,35	0,91	0,4	1	0,83	1	260	250	120,4
2	coba_2	183	0,41	1	0,8	2	0,9	3	243,9	250	333,3
3	coba_12	14.222	24	102	28	121	31	140	425	432,1	451,6
4	coba_13	15.225	26	109	30	130	34	150	419,2	433,3	441,1
5	coba_18	15.478	26	112	30	132	34	152	430,7	440	447,0
6	coba_14	17.984	30	129	35	154	40	177	430	440	442,5
7	coba_9	23.907	40	173	47	204	53	235	432,5	434,0	443,3
8	coba_5	28.160	47	203	55	241	62	278	431,9	438,1	448,3
9	coba_20	32.413	55	236	63	278	72	320	429,0	441,2	444,4
10	coba_10	35.572	64	274	74	324	83	373	428,1	437,8	449,3
11	coba_11	59.593	104	447	119	528	133	603	429,8	443,6	453,3
12	coba_6	76.800	137	596	159	696	177	797	435,0	437,7	450,2
13	coba_15	207.460	520	2.022	576	2.292	618	2.565	388,8	397,9	415,0
14	coba_4	214.528	559	2.132	604	2.405	649	2.681	381,3	398,1	413,0
15	coba_3	230.400	613	2.333	670	2.638	727	2.924	380,5	393,7	402,2

16	coba_16	239.614	657	2.454	728	2.773	766	3.067	373,5	380,9	400,3
17	coba_7	476.160	2.187	6.636	2.244	6.732	2.311	6.933	303,4	300	300
18	coba_22	518.161	2.321	6.963	2.378	7.134	2.445	7.335	300	300	300
19	coba_8	526.848	2.551	7.653	2.608	7.824	2.675	8.025	300	300	300

6. DAFTAR PUSTAKA

[1] J. Daemen and V. Rijmen, *Advanced Encryption Standard (AES)*, Federal Information Processing Standards Publication 197, November 26th 2001.

[2] V.Lusiana, *Perancangan Perangkat Lunak untuk Keamanan Informasi pada E-Mail dengan Menggunakan Algoritma AES dan RSA*, tesis, Magister Sistem Informasi Universitas Diponegoro, Semarang, 2009.

[3] R. Munir, *Kriptografi*, Penerbit Informatika, Bandung, 2006.

[4] B. Schneier, *Applied Cryptography, Protocols, Algorithms, and Source Code in C*, A John Wiley&Sons, 1996.

[5] B. Song, *Observations on the Cryptologic Properties of the AES Algorithm*, Disertasi, University of Wollongong Australia, 2004.

[6] W. Stallings, *Cryptography and Network Security Principles and Practice second edition*, Prentice Hall, 1999.

[7] R.D. Stinson, *Cryptography Theory and Practice*, Chapman & Hall / CRC USA, 2002.