Implementasi Kriptografi dengan Modifikasi Algoritma *Advanced Encryption Standard* (AES) untuk Pengamanan *File Document*

Lilik Asih Indrayani[1], I Made Suartana[2]

[1] Jurusan Teknik Informatika/Teknik Informatika, Universitas Negeri Surabaya
[2] Jurusan Teknik Informatika/Teknik Informatika, Universitas Negeri Surabaya

[1]lilikindrayani@mhs.unesa.ac.id
[2]madesuartana@unesa.ac.id

***Abstrak*— Algoritma AES (*Advanced Encryption Standard*) disebut algoritma dengan *cipher block symmetric* karena untuk memperoleh data yang telah dienkripsi menggunakan kunci rahasia atau *cipher key* yang sama ketika melakukan proses penyandian data (enkripsi). AES memiliki 3 kategori blok cipher: AES-128, AES-192, dan AES-256 dengan panjang kunci masing-masing 128 bit, 192 bit, dan 256 bit. Perbedaan dari ketiga urutan tersebut adalah panjang kunci yang mempengaruhi jumlah *round* (putaran). Pada penelitian ini, algoritma AES akan dimodifikasi dengan meningkatkan jumlah putaran bersamaan dengan panjang kunci menjadi 320 bit dengan 16 putaran dengan tujuan meningkatkan keamanan dari algoritma AES. Pengujian dilakukan dengan membandingkan waktu proses enkripsi dan dekripsi antara algoritma AES standar 10 putaran dengan algoritma AES modifikasi 16 putaran. File dokumen yang dapat dienkripsi hanya berupa file dengan format *pdf*, *docx*, dan *txt*. Hasil pengujian menunjukkan bahwa semakin besar putaran dan panjang kunci, maka semakin lama waktu yang digunakan dalam proses enkripsi maupun dekripsi. Hal ini dapat dibuktikan dengan algoritma AES modifikasi yang memiliki nilai waktu proses lebih besar dibanding algoritma AES standar sehingga dapat disimpulkan algoritma AES modifikasi memiliki tingkat keamanan yang lebih tinggi karena berpengaruh pada waktu yang dibutuhkan seorang kriptoanalis untuk memecahkan kode enkripsi.**

***Kata Kunci*— Kriptografi, AES (*Advanced Encryption* Standard), enkripsi, dekripsi, pengamanan file dokumen, modifikasi putaran AES**

I. PENDAHULUAN

Kini, komputer menjadi hal yang sangat berpengaruh bagi sarana distribusi data dan informasi. Di dalam suatu komputer, pastinya terdapat file atau dokumen yang berisi data-data yang bersifat umum maupun rahasia. Oleh karena itu, diperlukan pengamanan data dan informasi agar data tersebut hanya dapat dibuka atau diakses dan digunakan oleh si penerima atau orang yang memiliki kewenangan atas data tersebut. Data-data tersebut diamankan dengan tujuan agar pihak lain atau orang lain yang tidak memiliki hak akses tidak dapat mengetahui makna atau isi dari data tersebut.

Kriptografi merupakan bentuk solusi yang dapat ditawarkan di dalam keamanan komputer, karena yang menjadi pokok dari fungsi komputer adalah data ataupun informasi. Kriptografi memenuhi 2 syarat dari aspek keamanan informasi yaitu *confidentiality* (menjamin keamanan informasi data) dan *integrity* (proteksi terhadap informasi dari manipulasi data tanpa ijin yang berwenang). Kriptografi merupakan teknik untuk menyamarkan atau mengubah suatu informasi sehingga informasi tersebut ketika dikirim menjadi tidak bernilai/tidak memiliki makna. Misalkan seseorang yang ingin mengirim data rahasia/informasi dengan kata "barang", maka pada proses pengiriman kata yang berupa data rahasia tersebut akan diganti menjadi sesuatu yang tidak dapat dibaca misalkan "?%&#/|". Kata atau pesan yang tidak bermakna disebut ciphertext sedangkan kata atau pesan asli disebut plaintext.

Kriptografi terdiri dari kata *Crypto* dan *Grapho* dimana Crypto yang artinya rahasia (secret) dan Grapho yang artinya menulis (*writing*) dalam Bahasa Yunani. Kurniawan menulis sebuah buku yang ia beri judul "Kriptografi Keamanan Internet dan Jaringan Komunikasi", menjelaskan bahwa salah satu bentuk ilmu dan seni untuk melindungi keaslian atau keabsahan pesan yaitu melalui kriptografi (*Cryptography is the art and science of keeping message secure*) [1]. Sedangkan Sadikin menjabarkan bahwa kriptografi bukan lagi ilmu yang mempelajari teknik penyembunyian pesan saja, namun kriptografi juga mencakup teknik yang memberikan pengamanan data berlebih seperti halnya keutuhan data, kerahasiaan, otontikasi, dan sebagainya [2]. Kromodimoeljo mengemukakan bahwa kriptografi adalah suatu ilmu yang mengkaji teknik yang mengacak data menggunakan kunci sehingga menjadikan bentuk lain yang tidak mampu terbaca oleh manusia kecuali pihak yang mengetahui kuncinya [3]. Teknik tersebut dikenal dengan istilah enkripsi yaitu mengubah data polos (*plaintext*) menjadi bentuk yang tak terbaca (*ciphertext*).

Salah satu algoritma dalam kriptografi yaitu algoritma AES *Advanced Encryption Standard* (AES yang memanfaatkan teknik blok simetris dalam proses penyandian pesannya. Pengembang dari algoritma ini berasal dari Belgia yang bernama Dr. Vincent Rijmen dan Dr. Joan Daemen pada tahun 1997. Sebagai kandidat AES, mereka berdua mengajukan algoritma ini dan berhasil menjadikannya proposal terpilih bagi AES oleh NIST (*National Institute of Standard and Technology*) pada tanggal 26 November 2001.

Sebelum terpilihnya Rijndael sebagai algoritma yang paling tepat. Pada 26 Mei 2002 setelah Menteri Perdagangan menyetujui, AES menjadi standar pertama yang efektif bagi pemerintah Federal serta dapat diakses dan digunakan oleh

publik dengan sandi terbuka yang disetujui NSA untuk keamanan informasi. AES dapat bekerja pada beberapa network layer pada saat yang bersamaan sehingga metode enkripsi ini dapat bekerja dengan baik [4].

AES (*Advanced Encryption Standard*) dibuat untuk menggantikan standar enkripsi kriptografi yang lama dimana sudahh tidak terjamin lagi keamanannya yaitu DES (*Data Encryption Standard*). Algoritma AES disebut algoritma dengan *cipher block symmetric* karena untuk memperoleh data yang telah dienkripsi menggunakan kunci rahasia yang sama ketika melakukan proses penyandian data. Seperti contoh istilah AES-256 yang mengacu pada algoritma Rijndael dengan 256 bit sebagai panjang blok data serta 256 bit sebagai panjang kunci. AES memiliki 3 kategori blok cipher: AES-128, AES-192, dan AES-256 dengan panjang kunci masing-masing 128 bit, 19 2bit, dan 256 bit. Perbedaan dari ketiga urutan tersebut adalah panjang kunci yang mempengaruhi jumlah *round* (perputaran). AES-128 menggunakan 9 *round* utama dan 1 *final round* dengan total 10 putaran, AES-192 menggunakan 12 putaran dan AES-256 menggunakan 14 putaran. Algoritma AES mempunyai proses transformasi bytes dengan 4 jenis pada proses enkripsinya, yaitu: *AddRoundKey, SubBytes, ShiftRows dan MixColumns*. Sedangkan pada dekripsi, transformasi yang digunakan merupakan kebalikannya yaitu : *InvShiftRows, InvSubBytes, dan InvMixColumns.*

Enkripsi yaitu melakukan proses pengamanan data atau penyembunyian maupun proses mengkonversi pesan asli (*plaintext*) menjadi bentuk yang tidak jelas untuk dipahami bahkan dimengerti. Pada berbagai negara, enkripsi telah digunakan untuk keamanan informasi serta komunikasi, namun hanya digunakan untuk kepentingan mendesak akan kerahasiaan oleh organisasi atau individu tertentu

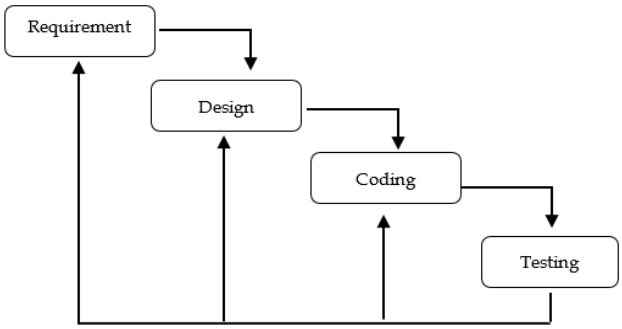
Kebalikan dari enkripsi, proses dekripsi merupakan proses mengkonversi/mengembalikan data rahasia atau data terenkripsi (*ciphertext*) menjadi plainteks (data asli) sehingga dapat dibaca atau kembali dimengerti.

Peningkatan keamanan dalam proses enkripsi/dekripsi dapat dilakukan dengan meningkatkan jumlah putaran/iterasi seperti pada penelitian oleh Puneet Kumar dan Shashi B. Rana [5] yang telah dilakukan sebelumnya dengan judul "*Development of Modified AES Algorithm for Data Security*". Penelitian tersebut memodifikasi AES standar menjadi 320 bit dengan 16 putaran untuk membuat sistem menjadi lebih aman. Sedangkan pada jurnal internasional lainnya yang berjudul "*AES Based Text Encryption Using 12 Rounds with Dynamic Key Selection*" oleh Rajesh Bansode dan Nishtha Mathur [6] menjelaskan bahwa untuk meningkatkan keamanan data maka algoritma AES standar dengan 128 bit dan 10 putaran ditingkatkan menjadi 192 bit dengan 12 putaran.

Dalam penelitian ini dikembangkan implementasi kriptografi dengan memodifikasi jumlah putaran pada algoritma AES (*Advanced Encryption Standard*) standar untuk tujuan pengamanan file dokumen yang berbasis data teks.

II. METODOLOGI PENELITIAN

Tahapan yang diterapkan dalam penelitian ini akan ditunjukkan pada gambar dibawah ini:



Gbr. 1 Metode Penelitian

Adapun tahapan yang dilakukan yaitu :

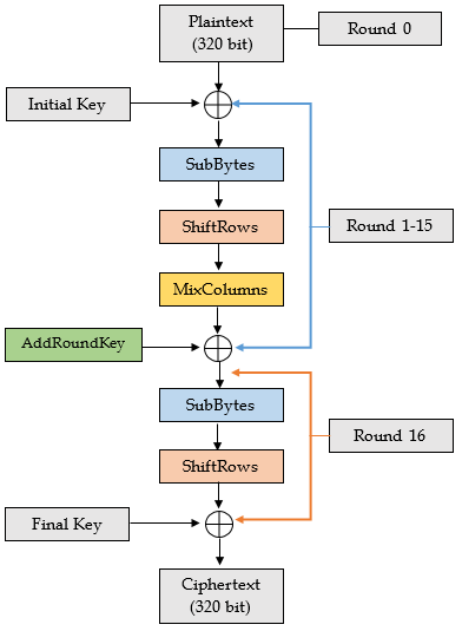
1. Analisa Kebutuhan (*Requirements*)
   Merupakan tahap menganalisa kebutuhan apa saja yang digunakan atau diperlukan dalam merancang sistem yang akan dibangun.

2. Desain (*Design*)
   Pada tahap ini data yang dianalisa terlebih dahulu diterjemahkan ke dalam bentuk yang mudah dipahami oleh pengguna (user). Pada tahap ini menampilkan desain perangkat lunak yang akan dibangun seperti menampilkan alur kerja sistem sehingga memberikan gambaran bagaimana dan seperti apa alur yang ada pada sistem.

3. Implementasi/Pengkodean (*Coding*)
   Tahap ini merupakan implementasi dari perancangan alur kerja sistem dengan menggunakan bahasa pemrograman tertentu untuk menghasilkan aplikasi yang telah dirancang/didesain. Dalam hal ini, sistem yang telah didesain tersebut nantinya akan diimplementasikan dalam bentuk aplikasi berbasis desktop.

4. Pengujian (*Testing*)
   Pada tahap ini dimana sistem telah melalui tahap pengkodean lalu selanjutya masuk ke tahap pengujian. Tahap pengujian ini dilakukan terhadap perangkat lunak yang dibangun untuk membandingkan tingkat keamanan pada sistem yang telah dibangun dengan sistem standar yang sudah ada sebelumnya.

Alur sistem yang dibangun pada penelitian adalah dengan menambahkan putaran (*round*) menjadi 16 putaran dan panjang kunci menjadi 320 bit. Untuk lebih jelasnya akan ditunjukkan pada Gbr 2 berikut ini :

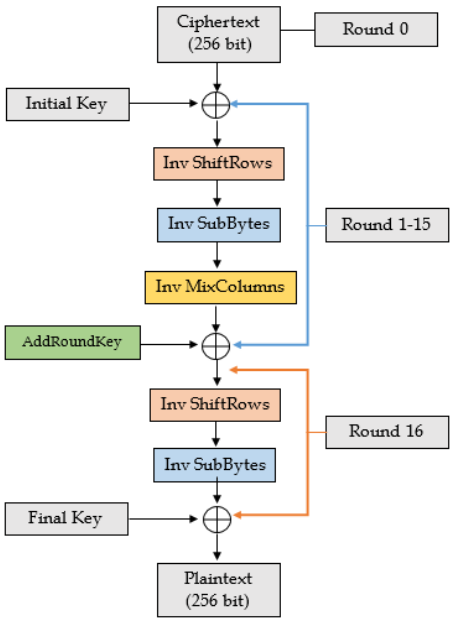


Gbr. 2 Proses Enkripsi AES Modifikasi

Alur proses enkripsi dari AES modifikasi adalah sebagai berikut :

1. Data masuk atau data yang telah diupload kemudian ditampung ke dalam blok data yang berbentuk array 2 dimensi yang bernama state array. Data teks yang masuk dikonversikan terlebih dahulu ke dalam bentuk bit menggunakan kode ASCII. Lalu dari kode ASCII dikonversikan ke dalam heksadesimal.
2. Kemudian state array ini di XOR kan dengan cipher key yang telah dimasukkan dan dikonversi ke dalam bentuk array kunci. Cipher key yang pertama diberi nama initial key. Sedangkan proses ini dinamakan Addroundkey.
3. Selanjutnya proses substitusi bytes (SubBytes) yaitu hasil dari XOR disubstitusikan dengan tabel Rijndael S-Box.
4. Kemudian dilakukan proses ShiftRows yaitu pergeseran wrapping pada setiap elemen blok yang bekerja pada setiap baris dimana baris pertama tidak mengalami pergeseran, baris ke-2 melakukan pergeseran 1 byte, baris ke-3 melakukan pergeseran 2 byte, dan baris ke-4 melakukan pergeseran 3 byte.
5. Setelah itu transformasi MixColumns yaitu masing-masing elemen dari cipher block dikalikan dengan matriks yang telah ada. Pengalian yang dilakukan menggunakan dot product sama halnya perkalian matriks biasa, kemudian hasil perkalian keduanya ditampung ke dalam sebuah cipher block baru.
6. Setelah MixColumns selesai, maka dilakukan AddRoundKey yaitu melakukan XOR state array dengan round key.
7. Setelah AddRoundKey selesai maka kembali ke proses SubBytes dan seterusnya hingga melalui 15 kali putaran.
8. Pada putaran terakhir atau putaran ke-16, setelah melakukan ShiftRows tidak perlu melakukan proses MixColumns namun langsung pada proses XOR antara hasil state array dengan array kunci terakhir atau final key.
9. Proses terakhir yaitu data yang keluar atau data output berupa ciphertext.

Selanjutnya mengenai alur proses dekripsi ditunjukkan pada Gbr 3 berikut ini :



Gbr. 3 Proses Dekripsi AES Modifikasi

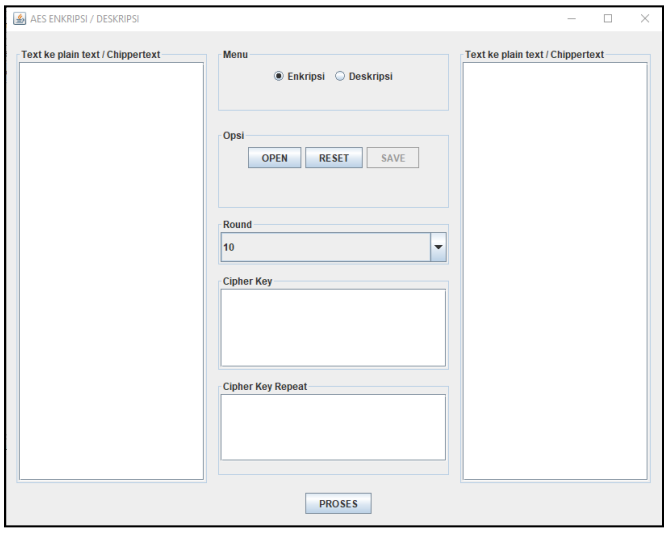
Alur proses dekripsi dari AES modifikasi adalah sebagai berikut :

1. Data yang telah dienkripsi (ciphertext file) mengalami transformasi AddRoundKey yaitu pencampuran antara kunci ronde ke-16 dengan state atau blok array yang berisi ciphertext hasil enkripsi dengan perhitungan XOR.
2. Setelah transformasi AddRoundKey selesai maka selanjutnya adalah transformasi Inverse Shiftrows dimana merupakan kebalikan dari transformasi Shiftrows.
3. Kemudian melakukan transformasi Inverse SubBytes dimana merupakan transformasi yang terakhir pada putaran pertama atau pra-ronde. Pada transformasi ini dilakukan proses substitusi terhadap state hasil transformasi yang sebelumnya menggunakan tabel Inverse SubBytes.
4. Selanjutnya masuk pada putaran berikutnya atau ronde ke-1 dimana terjadi transformasi AddRoundKey yaitu pencampuran antara kunci ronde ke-15 dengan state hasil putaran sebelumnya.
5. Kemudian transformasi selanjutnya yaitu Inverse MixColumns yaitu melakukan perkalian matriks antara state hasil AddRoundKey dengan matriks yang telah tersedia untuk proses dekripsi.
6. Setelah transformasi Inverse MixColumns selesai, selanjutnya melakukan transformasi Inverse ShiftRows dan Inverse SubBytes.

7. Langkah 4-6 diulang pada ronde berikutnya yaitu ronde ke-2 sampai ronde ke-15, tetapi ada perbedaan terkait penggunaan kunci ronde pada transformasi AddRoundKey dimana ronde ke-1 sampai ke-15 menggunakan kunci ronde ke-15 sampai ke-1.
8. Pada putaran terakhir, transformasi yang dilakukan hanya AddRoundKey dimana state hasil ronde ke-15 dicampur dengan kunci ronde ke-0 atau pra ronde yang biasa disebut kunci cipher.
9. Setelah itu, proses berakhir dengan plaintext atau teks asli akan didapatkan

III. HASIL DAN PEMBAHASAN

A. *Implementasi Program*

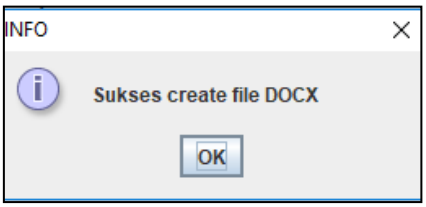


Gbr. 4 Tampilan Utama Program

Gbr 4 merupakan tampilan utama dari program yang telah dibuat dimana di sebelah kiri dan kanan terdapat kolom yang digunakan untuk menampilkan informasi file enkripsi atau dekripsi yang telah dilakukan. Kemudian ada panel menu yang berisi 2 pilihan yaitu enkripsi dan dekripsi. Selanjutnya ada panel opsi yang berisi 3 pilihan yaitu *open*, *reset*, dan *save*. Lalu ada *combobox round* untuk pilihan panjang kunci yang diinginkan. Kemudian kolom *cipher key* untuk memasukkan kunci enkripsi/dekripsi. Dan yang terakhir kolom *cipher key repeat*. Namun kolom ini tidak bisa diisi oleh pengguna, karena sistem yang akan mengisi secara otomatis sesuai kebutuhan.

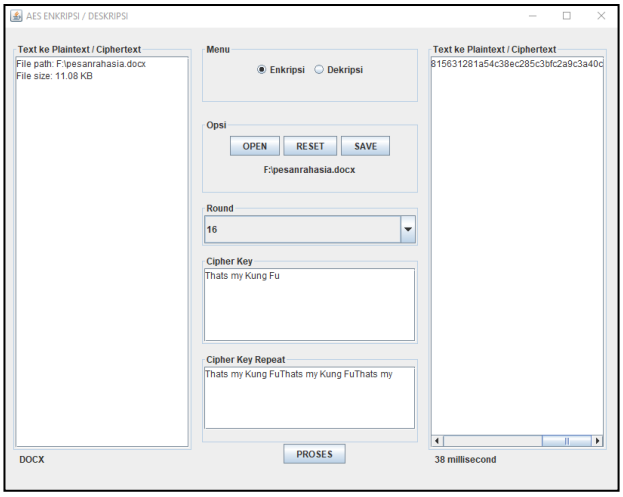
Ada enam langkah yang yang harus dilakukan pengguna untuk melakukan proses enkripsi maupun dekripsi :

1. Pengguna harus memilih diantara dua pilihan yang tersedia, yaitu enkripsi atau dekripsi.
2. Pengguna memilih file dokumen yang akan dienkripsi atau didekripsi melalui tombol *Open* yang tersedia, namun pengguna hanya dapat memilih file dokumen dengan format *docx*, *pdf*, dan *txt*.
3. Pengguna memilih besar *round* atau putaran yang akan digunakan untuk proses enkripsi maupun dekripsi.
4. Selanjutnya pengguna memasukkan *cipher key* atau kunci yang digunakan dalam melakukan proses enkripsi atau dekripsi. Kemudian kolom *cipher key repeat* akan otomatis terpenuhi sesuai dengan besar *round* yang dipilih.
5. Pengguna mengklik tombol *Proses* yang kemudian muncul pemberitahuan bahwa file telah berhasil dienkripsi atau dekripsi.
6. Pengguna mengklik tombol *Save* untuk menyimpan hasil dari proses enkripsi atau dekripsi yang telah dilakukan. Kemudian klik tombol *Ok* pada pop up untuk menutup *message box* dan proses enkripsi atau dekripsi telah selesai. (*Lihat gambar 5*)

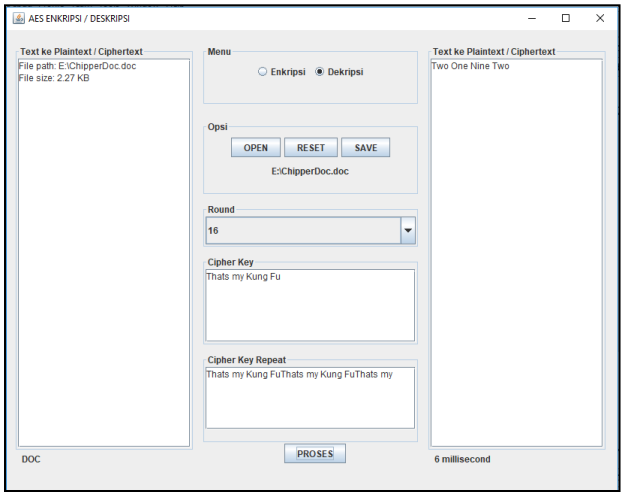


Gbr. 5 Notifikasi Simpan Berhasil

Adapun tampilan ketika proses enkripsi dan dekripsi dilakukan ditunjukkan pada Gbr 6 dan 7 berikut ini :



Gbr. 6 Tampilan Proses Enkripsi Selesai



Gbr. 7 Tampilan Proses Dekripsi Selesai

Ketika proses enkripsi dan dekripsi selesai, maka pada kolom sebelah kanan (*gambar 6 dan 7*) akan menampilkan hasil *encrypt/decrypt* yang telah dilakukan. Berbeda dengan kolom sebelah kanan, pada kolom sebelah kiri hanya akan menampilkan informasi terkait nama dan ukuran file yang telah diinputkan. Kemudian di bagian bawah sebelah kanan terdapat informasi waktu dari proses enkripsi/dekripsi yang telah dilakukan.

B. *Hasil Pengujian*

Proses pengujian meliputi *computation time* atau waktu yang diperlukan dalam melakukan proses enkripsi dan dekripsi dengan beberapa macam format file dokumen dan ukuran file yang berbeda. Pengujian dilakukan terhadap tiga jenis file dokumen yakni *.pdf, .docx, .txt*. Pengujian dilakukan dengan membandingkan antara algoritma AES standar 128bit 10 putaran dengan sistem yang dibangun yaitu AES 320bit 16 putaran. Waktu adalah salah satu variabel pengujian yang digunakan untuk menentukan apakah sistem yang dibangun dapat berjalan lebih baik dari sistem yang telah ada sebelumnya.

Hasil pengujian akan ditunjukkan pada tabel berikut ini :

TABEL I
HASIL PENGUJIAN AES 10 PUTARAN

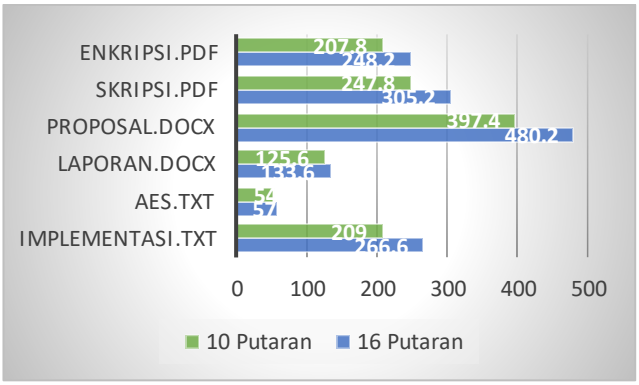
No	Nama	*Size* (MB)	Waktu Enkripsi (milidetik)	Waktu Dekripsi (milidetik)
1	Enkripsi.pdf	0.53	207.8	319.4
2	Skripsi.pdf	3.13	247.8	333.6
3	Proposal.docx	0.8	397.4	580.6
4	Laporan.docx	31.8	125.6	170.6
5	AES.txt	0.01	54.2	155.4
6	Implementasi.txt	0.11	209	391.2

TABEL III
HASIL PENGUJIAN AES 16 PUTARAN

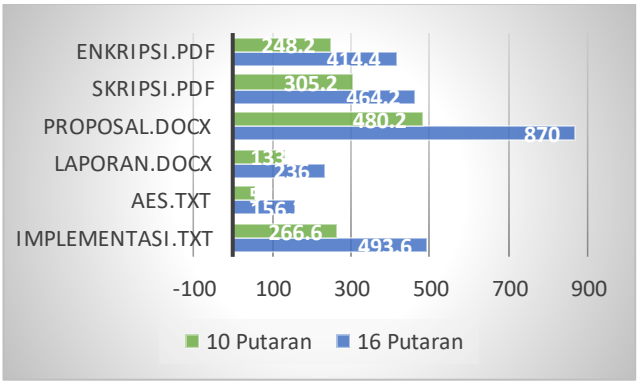
No	Nama	*Size* (MB)	Waktu Enkripsi (milidetik)	Waktu Dekripsi (milidetik)
1	Enkripsi.pdf	0.53	248.2	414.4
2	Skripsi.pdf	3.13	305.2	464.2
3	Proposal.docx	0.8	480.2	870.0
4	Laporan.docx	31.8	133.6	236.0
5	AES.txt	0.01	57.0	156.2
6	Implementasi.txt	0.11	266.6	493.6

Hasil pengujian dilakukan pada enam file dokumen yang memiliki tiga format file berbeda yaitu file pdf, docx dan txt. Pengujian dilakukan sebanyak 5x untuk masing-masing file dokumen dengan *cipher key* yang sama. Waktu yang diperoleh dari 5x pengujian pun sangat bervariasi, maka dari itu kemudian diambil rata-rata waktu yang hasilnya terdapat pada tabel I dan II diatas.

Untuk lebih jelasnya perbandingan antara AES modifikasi yang telah dibangun dengan AES standard yang telah ada akan ditunjukkan melalui Gbr 8 dan 9 dibawah ini :



Gbr. 8 Grafik Perbandingan Waktu Enkripsi



Gbr. 9 Grafik Perbandingan Waktu Dekripsi

Grafik pada gambar 8 dan 9 menunjukkan untuk variabel sebelah kiri adalah nama file yang diuji coba, kemudian bagian bawah yaitu *range* nilai yang digunakan.

Berdasarkan grafik pada gambar 8 dan 9, proses enkripsi dan dekripsi dengan menggunakan algoritma AES modifikasi lebih lama dibandingkan algoritma AES standar. Hal ini menujukkan bahwa dengan menggunakan algoritma AES modifikasi akan membuat peretas semakin sulit karena membutuhkan waktu yang lebih lama lagi.

IV. KESIMPULAN

Berdasarkan hasil penelitian yang dilakukan maka dapat diambil kesimpulan bahwa modifikasi algoritma AES untuk proses enkripsi dan dekripsi file dokumen dengan menambah jumlah putaran dan panjang kunci memiliki pengaruh terhadap lamanya waktu proses enkripsi dan dekripsi. Semakin besar putaran dan panjang kunci, maka semakin lama waktu yang digunakan dalam proses enkripsi maupun dekripsi. Serta dari hasil pengujian berdasarkan perbandingan *computation time* antara algoritma AES standar dengan algoritma AES modifikasi menunjukkan bahwa algoritma AES modifikasi memiliki nilai waktu yang lebih besar dibanding algoritma AES standar sehingga dapat dikatakan algoritma AES modifikasi memiliki tingkat keamanan yang lebih tinggi karena berpengaruh pada waktu yang dibutuhkan seorang kriptoanalis untuk menghack sistem akan lebih lama lagi.

UCAPAN TERIMA KASIH

Ucapan terima kasih kepada Allah SWT yang selalu memberikan kemudahan dan kelancaran sehingga jurnal ini dapat terselesaikan dengan baik.

REFERENSI

[1] Kurniawan, Y. (2004). Keamanan Internet dan Jaringan Telekomunikasi. Bandung: Informatika.
[2] Sadikin, R. (2012). Kriptografi untuk Keamanan Jaringan dan Implementasinya dalam Bahasa Java. Yogyakarta: CV. Andi Offset.
[3] Kromodimoeljo, S. (2010). Teori & Aplikasi Kriptografi. Jakarta: SPK IT Consulting.
[4] Daemen, J., & Rijmen, V. (n.d.). Computer Security Resource Center. Retrieved from NIST: https://csrc.nist.gov/csrc/media/projects/cryptographic-standards-and-guidelines/documents/aes-development/rijndael-ammended.pdf
[5] Kumaar, P., & Rana, B. S. (2015). Development of Modified AES Algorithm for Data Security. Jurnal Optik, 2341-2345.
[6] Mathur, N., & Bansode, R. (2016). AES Based Text Encryption Using 12 Rounds with Dynamic Key Selection. Procedia Computer Science (pp. 1036-1043). India: Elsevier B.V.