[4] Erizal. 2013. *Prototipe* Sistem *Pendukung Keputusan Dengan Menerapkan Logika Fuzzy Untuk Penilaian Kinerja Dosen.* Jakarta : Universitas Budi *Luhur*.

[5] Frans Susilo .2006. *Himpunan dan Logika Kabur serta aplikasinya*. Jakarta : Graha Ilmu

[6] Andrews, Keith. 2013. *Human Computer Interaction.*Infelldagasse: Graz University of Technology.

[7] Syarifullah, Lutfi. 2013. *Kajian Penerapan* ANFIS Dalam Penentuan Beasiswa. *Jakarta* : Universitas Budi Luhur.

[8] [Marimin *2013*] Marimin dan Nurul Magfiroh. 2013. *Aplikasi Teknik Pengambilan Keputusan Dalam Manajemen Rantai Pasok.* Bogor : IPB Press

[9] Moedjiono. 2012. *Pedoman Penelitian, Penyusunan dan Penilaian Tesis (V.5). www.budiluhur.ac.id*. Jakarta : Universitas Budi Luhur.

[10] Prabowo Pudjo Widodo dan Rahmadya Trias Handayanto. 2012. *Penerapan Soft Computing dengan Matlab*. Bandung. Rekayasa Sains.

[11] Kusumadewi, Sri. 2003. *Artificial Intelegence (Teknik dan Aplikasinya).* Yogyakarta. Graha Ilmu.

[12] Kadarsyah Suryadi dan Ali Ramdhani. 1998. *Sistem pendukung keputusan*. Bandung: Remaja Rosdakarya

[13] Sutojo. 2010. *Kecerdasan Buatan. Semarang* : Andi Yogyakarta.

[14] [Wahyudi 2009] Sri Herawati dan Wahyudi Agustiono. 2009. *Interaksi Manusia dan Komputer*. Bangkalan. ITS.

IMPLEMENTASI KEAMANAN DATA DENGAN ALGORITMA KUNCI SIMETRIS RIJNDAEL MENGGUNAKAN VB.NET 2008

AMAT SUROSO
Program Studi Manajemen Informatika
STMIK Bani Saleh
Email : ahmad_suroso04@yahoo.com

ABSTRAK

Seiring dengan perkembangan zaman, kebutuhan manusia meningkat. Termasuk kebutuhan akan informasi. Oleh sebab itu, pengiriman dan penyimpanan data melalui media elektronik memerlukan suatu proses yang mampu menjamin keamanan dan keutuhan dari data tersebut. Untuk menjamin keamanan dan keutuhan dari suatu data, dibutuhkan suatu proses penyandian.

Enkripsi dilakukan ketika data akan dikirim. Proses ini akan mengubah suatu data asal menjadi data rahasia yang tidak dapat dibaca. Sementara itu, proses dekripsi dilakukan oleh penerima data yang dikirim tersebut. Data rahasia yang diterima akan diubah kembali menjadi data asal. Dengan cara penyandian tadi, data asli tidak akan terbaca oleh pihak yang tidak berkepentingan, melainkan hanya oleh penerima yang memiliki kunci dekripsi.

Sistem keamanan yang akan dibahas adalah sistem keamanan dengan menggunakan algoritma Rijndael. Algoritma Rijndael dapat mengenkripsi dan mendekripsi blok data sepanjang 128 bit dengan panjang kunci 128 bit, 192 bit, atau 256 bit. Semakin besar bit yang digunakan maka data yang dienkripsi akan menjadi semakin sulit untuk bisa dimodifikasi ataupun diambil oleh orang yang tidak berkepentingan. Suatu aplikasi enkripsi harus memperhatikan lamanya waktu yang dibutuhkan untuk melakukan suatu proses enkripsi sehingga aplikasi tersebut dapat menjadi aplikasi yang baik.

Kata kunci : enkripsi, dekripsi, rijndael.

Abstrack

Along with the development of the times, human needs increase. Includes the need for information. Therefore, the transmission and storage of data through electronic media requires a process that is able to ensure the security and integrity of the data. To ensure the security and integrity of a data, an encoding process is required.

Encryption is done when data is sent. This process will convert the origin data into confidential data that can not be read. Meanwhile, the decryption process is performed by the recipient of the transmitted data. Confidential data received will be changed back into the original data. By way of encoding, the original data will not be read by unauthorized parties, but only by the recipient who has the decryption key.

The security system to be discussed is the security system using Rijndael algorithm. The Rijndael algorithm can encrypt and decrypt 128-bit data blocks with 128 bit, 192 bit, or 256 bit key lengths. The larger the bits used, the encrypted data becomes more difficult to modify or be taken by unauthorized people. An encryption application should pay attention to the length of time it takes to perform an encryption process so that the application can be a good application.

Keywords: encryption, decryption, rijndael.

1. PENDAHULUAN

Keamanan dan kerahasiaan data merupakan hal yang sangat penting dalam suatu organisasi maupun pribadi jika data yang berada dalam suatu jaringan komputer terhubung dengan jaringan lain. Hubungan tersebut tentu saja akan menimbulkan resiko jika informasi yang sensitif dan berharga tersebut diakses oleh orang-orang yang tidak berhak. Pengaksesan data yang bersifat rahasia kemungkinan besar akan merugikan bahkan membahayakan orang yang mengirim pesan atau menerima pesan, maupun organisasinya.

beberapa cara yang dapat kita lakukan untuk mengamankan data supaya data tersebut terjaga kerahasiaannya antara lain dengan memberikan password untuk bisa mengakses data tersebut, menggunakan *internet firewall* a*tau secure socket layer* apabila kita menggunakan jaringan internet untuk mengakses data tersebut,dan salah satu cara yang banyak digunakan saat ini adalah dengan menggunakan kriptografi.

Kriptografi merupakan salah satu metode untuk mengamankan data yaitu dengan mengenkripsi data / pesan aslinya dan untuk bisa membaca data / pesan tersebut, maka harus dilakukan proses dekripsi. Enkripsi adalah suatu cara untuk menyandikan suatu informasi menjadi sebuah kode – kode rahasia, sedangkan dekripsi adalah suatu cara untuk mengubah kode – kode rahasia tadi menjadi informasi dengan menggunakan kunci rahasia. Beberapa algoritma untuk mengenkripsi data / pesan antara lain : *Data Encryption Standart* (*DES*) , *Blowfish, Twofish*, International Data Encryption Algorithm(*IDEA), MARS, 3DES(DES* diaplikasikan 3 kali*), RSA (Rivest-Shamir-Adleman), Knapsack, Rijndael, Message Digest Algorithm-5(MD-5)* dan masih banyak algoritma yang lainnya.

Dalam tugas akhir ini penulis akan mengimplementasikan salah satu metode kriptografi dengan mengunakan algoritma Rijndael. Algoritma Rijndael dapat mengenkripsi dan mendekripsi blok data sepanjang 128 bit dengan panjang kunci 128 bit, 192 bit, atau 256 bit.

6 2. METODOLOGI PENELITIAN

6.1 2.1. Keamanan Komputer / Data

Dalam dunia komunikasi global dan perkembangan teknologi informasi yang senantiasa berubah serta cepatnya perkembangan software, keamanan merupakan suatu isu yang sangat penting, baik itu keamanan fisik, keamanan data maupun keamanan aplikasi.

Pengertian keamanan komputer menurut beberapa ahli antara lain :

a. Menurut John D.Howard dalam bukunya "*An analysis of security incidents on the internet* " menyatakan bahwa keamanan komputer adalah tindakan pencegahan dari serangan pengguna komputer atau pengakses jaringan yang tidak bertanggung jawab.
b. Menurut Gollman pada tahun 1999 dalam bukunya "*Computer Security*" menyatakan bahwa keamanan komputer adalah dengan pencegahan diri dan deteksi terhadap tindakan pengganggu yang tidak dikenali dalam sistem komputer.

Ada beberapa hal yang bisa menjawab mengapa kita perlu mengamankan sistem komputer , antara lain :

1. Menghindari resiko penyusup, kita harus memastikan bahwa sistem tidak kemasukkan penyusup yang bisa membaca, menulis, dan menjalankan program – program yang bisa menghancurkan sistem kita.
2. Mengurangi resiko ancaman, hal ini biasa berlaku di institusi dan perusahaan swasta.
3. Melindungi sistem dari kerentanan, kerentanan akan menjadikan sistem kita berpotensi untuk memberikan akses yang tidak diizinkan bagi orang lain yang tidak berhak.
4. Melindungi sistem dari gangguan alam, seperti petir dan lain – lainnya.

6.1.1 2.2. Pengertian Kriptografi

Kriptografi merupakan seni dan ilmu menyembunyikan informasi dari penerima yang tidak berhak. Kata cryptographi berasal dari bahasa yunani yaitu kryptos (tersembunyi) dan graphein (menulis). *Criptanalysis* adalah aksi untuk memecahkan mekanisme kriptografi dengan cara mendapatkan plaintext atau kunci dari ciphertext yang digunakan untuk mendapatkan informasi berharga kemudian mengubah atau memalsukan pesan dengan tujuan untuk menipu penerima yang sesungguhnya. Enkripsi adalah mentransformasi data kedalam bentuk yang tidak dapat terbaca tanpa sebuah kunci tertentu. Tujuannya adalah untuk meyakinkan privasi dengan menyembunyikan informasi dari orang – orang yang tidak ditujukan, bahkan dari mereka yang memiliki akses ke data terenkripsi. Deskripsi merupakan kebalikan dari enkripsi yaitu transformasi data terenkripsi kembali ke bentuk semula.

Kriptografi adalah seni dan ilmu untuk menjaga agar pesan rahasia tetap aman (Schneier, 1996). Kriptografi adalah salah satu cabang ilmu algoritma matematika. Ada dua tipe dasar dari teknologi kriptografi yaitu

symmetric key (secret / private key) cryptography dan asymmetric key (public key) cryptography. Pada *symmetric key cryptography* baik pengirim maupun penerima memilki kunci rahasia yang umum. Pada *asymetric key cryptography* pengirim dan penerima masing – masing berbagi kunci public dan privat.

6.1.2 2.3. Macam – Macam Metode Kriptografi

6.1.2.1 2.3.1. *Substitusi*

Caesar *cipher* adalah *cipher* substitusi sederhana yang mencakup pergeseran alfabet 3 posisi ke kanan. Caesar *cipher* merupakan *subset* dari *cipher* polialfabetik vigenere. Pada Caesar *cipher* karakter – karakter dan pengulangan kunci dijumlahkan bersama, modulo 26. Dalam penjumlahan modulo 26 huruf – huruf A – Z dari alfabet masing – masing memberikan nilai 0 sampai 25. Tipe *cipher* ini dapat diserang dengan menggunakan analisis frekuensi. Dalam frekuensi analisis digunakan karakteristik frekuensi yang tampak dalam penggunaan huruf – huruf alfabet pada bahasa tertentu. Tipe *crytanalysis* ini dimungkinkan karena Caesar cipher adalah monoalfabetik *cipher* atau *cipher* substitusi sederhana, dimana karakter *ciphertext* disubtitusi untuk setiap karakter *plaintext*. Serangan ini dapat diatasi dengan menggunakan substitusi polialfabetik. Substitusi polialfabetik dicapai melalui penggunaan beberapa *cipher* substitusi, namun substitusi ini dapat diserang dengan penemuan periode, saat substitusi berulang kembali (Hartono, 2007).

6.1.2.2 **2.3.2. *Transposisi* (Permutasi)**

Permutasi adalah memindahkan atau merotasikan karakter dengan aturan tertentu. Sebagai contoh : huruf – huruf plaintext A T T A C K A T D A W N dapat dipermutasi jadi D C K A A W N A T A T T . *Cipher transposisi* kolumnar adalah *cipher* dimana *plaintext* ditulis secara horizontal pada kertas dan dibaca secara vertikal. *Cipher* dapat diserang melalui analisis frekuensi, namun *cipher* menyembunyikan properti statistik dari pasangan huruf – huruf seperti IS dan TOO (Hartono, 2007).

6.1.2.3 2.3.3. *Vernam Cipher (One Time Pad)*

Cipher ini diimplementasikan melalui sebuah kunci yang terdiri dari sekumpulan *random* karakter – karakter yang tidak berulang. Setiap huruf kunci dijumlahkan modulo 26 dengan huruf *plainext*. Pada *One Time Pad* tiap huruf kunci digunakan satu kali untuk satu pesan dan tidak digunakan kembali. Panjang *stream* karakter kunci sama dengan panjang pesan (Hartono, 2007).

6.1.2.4 **2.3.4. *Book Key Cipher / Running Key Cipher***

Cipher ini menggunakan teks dari sebuah sumber (misalnya buku) untuk mengenkripsi *plaintext*. Kunci diketahui oleh pengirim dan penerima yang dimaksud dapat berupa halaman dan jumlah baris dari teks pada buku. Teks ini adalah karakter yang sesuai untuk karakter dengan *plaintext*, dan penjumlahan modulo 26 dijalankan untuk mempengaruhi enkripsi. *Running key cipher* mengeliminasi priodisitas, namun masih dapat diserang dengan memanfaatkan redudansi pada kunci (Hartono, 2007).

6.1.2.5 2.3.5. *Codes*

Codes berkaitan dengan kata – kata dan frase dan menghubungkan kata – kata ini sebagai frase untuk sekelompok angka atau huruf. Sebagai contoh angka 526 dapat berarti *"Attack at dawn"* (Hartono, 2007).

2.3.6. Steganography

Adalah seni menyembunyikan keberadaan pesan. "*Steganography*" berasal dari kata Yunani "steganos" yang berarti "terlindungi" dan "graphein" yamg berarti "menulis". Sebuah contohnya adalah *microdot* yang mengkompresi pesan kedalam ukuran period atau dot. Steganography dapat digunakan untuk membuat *"watermark"* digital untuk mendeteksi penyalinan image digital secara illegal (Hartono, 2007).

2.3.7. Algoritma Rijndael sebagai *AES*

Data Encryption Standard (DES) merupakan algoritma yang teraman didunia selama puluhan tahun, namun masih memiliki kekurangan yaitu pada panjang bit dari *DES* hanya 56 bit, sehingga dianggap terlalu pendek. Karena didalam algoritma kriptografi modern dengan penggunaan komputer yang sangat intensif, panjang ukuran bit yang digunakan sebagai kunci sangat berpengaruh. Untuk mengatasi hal itu , maka *NIST* mempersiapkan algoritma pengganti *DES*, yang disebut *Advance Encryption Standard (AES)*. Kontes terbuka untuk mendapatkan AES dimukai pada tahun 1997 dengan jumlah peserta sebanyak 21 tim. Pada seleksi tahap satu enam algoritma gugur, karena dinilai tidak sesuai dengan kriteria. Seleksi tahap dua menggugurkan 10 dari 15 algoritma lainnya yang dianggap kurang aman ataupun kurang efisien untuk diimplementasikan. Setelah terpilih lima kandidat akhirnya pada tahun 2000 terpilih sebuah algoritma *AES* yang juga dikenal dengan nama Rijndael, sesuai dengan nama penciptanya yaitu Dr. Vincent Rijmen dan Dr. Joan Daemen. Alasan terpilihnya Rijndael adalah karena algoritma tersebut memiliki keseimbangan antara keamanan dan fleksibilitas dalam berbagai *platform* baik *software* maupun *hardware*. Selain itu kesederhanaan dari rancangan algoritma ini membuatnya memakan waktu yang lebih singkat, bila dibandingkan dengan kandidat – kandidat pesaingnya (Hartono, 2007).

Disesuaikan dengan fleksibilitas panjang ukuran kunci yang diinginkan, Rijndael menyusun kombinasi berikut untuk kunci blok ronde :

TABEL 2.1 Kombinasi Panjang Kunci, Ukuran Blok, dan Jumlah Putaran

	Panjang kunci Nk Words	Ukuran blok Nb Words	Ronde (rounds) Nr
AES-128 Bit	4	4	10
AES-192 bit	6	4	12
AES-256 bit	8	4	14

Sumber : (Munir, 2006)

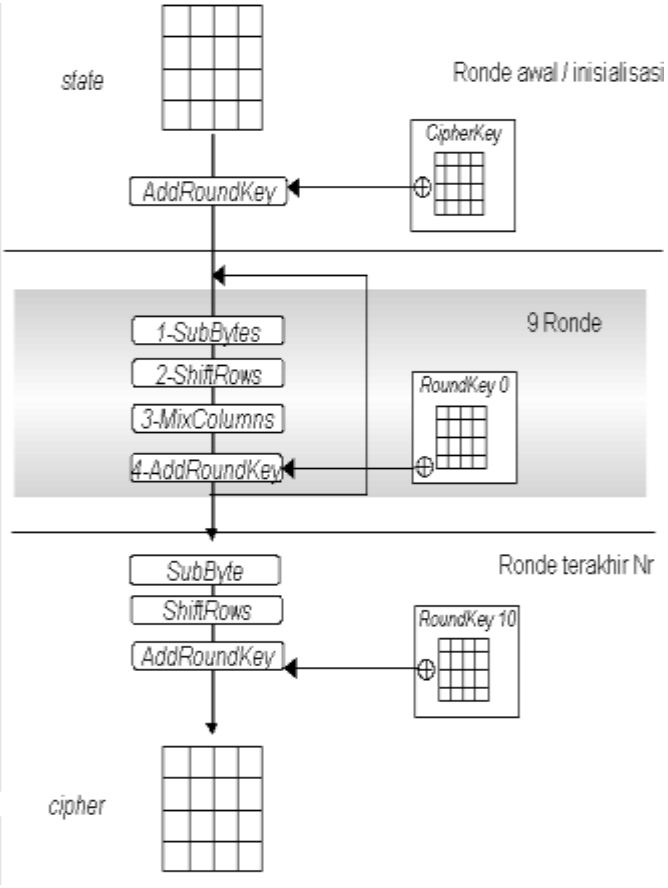
2.3.8. Enkripsi Rijndael

Proses enkripsi pada algoritma AES terdiri dari 4 jenis transformasi *bytes*, yaitu *subByte*, *shiftRows*, *Mixcolumns*, dan *AddRoundKey*. Pada awal proses enkripsi, input yang telah dikopikan ke dalam *state* akan mengalami transformasi *bytes addRoundKey*. Setelah itu, state akan mengalami transformasi *SubBytes*,*Shiftrows*, *Mixcolumns*, dan *AddRoundKey* secara berulang – ulang sebanyak *Nr*. Proses ini dalam algoritma AES disebut sebagai *round function*. *Round* yang terakhir agak berbeda dengan *round – round* sebelumnya di mana pada *round* terakhir *state* tidak mengalami transformasi *Mixcolumns*.

Operasi enkripsi Rijndael dapat dinyatakan dengan kode semu (pseudocode) berikut ini (Kurniawan, 2004) :

Pseudocode Cipher Rijndael :

```
Cipher(byte in[], byte out[], word w[] /*Nama fungsi*/
Begin
In = 4*Nb
Out = 4*Nb
W = Nb*(Nr+1)
Byte state[4,Nb]
State = In /*memasukkan input ke state*/
AddRoundKey(state,w)
For round = 1 step 1 to Nr-1 /*proses yang berlaku untuk semua ronde  kecuali ronde terakhir*/
SubBytes(state) /*proses yang berlaku untuk ronde terakhir*/
ShiftRows(state)
AddRoundKey(state,w+round*Nb) /*mengirim keluaran ke out*/
Out = state
End
```



Sumber : (Vony Yuniati, 2009)
GAMBAR 2.3 Diagram Alir Proses Enkripsi

a. SubBytes

SubByte merupakan transformasi *byte* dimana setiap elemen pada *state* akan dipetakan dengan menggunakan sebuah *table substitusi* (S-Box). Hasil yang didapat dari pemetaan dengan menggunakan *table* S-Box ini sebenarnya adalah hasil dari dua proses transformasi *byte*, yaitu :

1. *Invers* perkalian dalam GF(2^8) adalah fungsi yang memetakan 8 bit ke 8 bit yang merupakan *invers* dari elemen *finite field* tersebut. Suatu *byte* a merupakan *invers* perkalian dari *byte* b bila a.b = 1, kecuali {00} dipetakan ke dirinya sendiri. Setiap elemen pada state akan dipetakan pada *table invers*. Sebagai contoh : elemen "01010011" atau {53} akan dipetakan ke {CA} atau "11001010"
2. *Transformasi affine* pada *state* yang telah dipetakan. Transformasi *affine* ini apabila dipetakan dalam bentuk matriks adalah sebagai berikut :

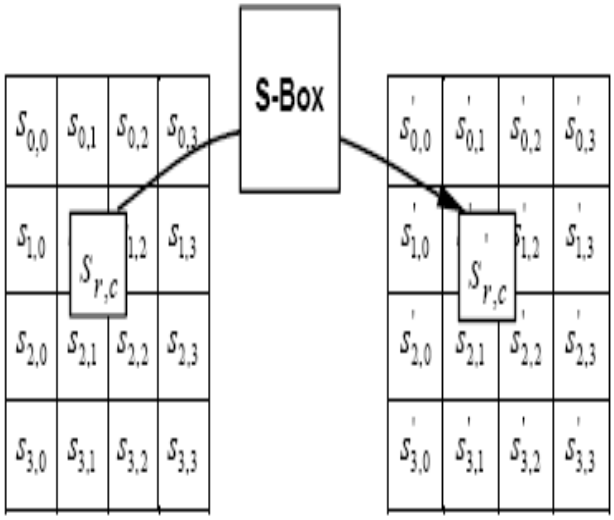
$$\begin{bmatrix} b_0' \\ b_1' \\ b_2' \\ b_3' \\ b_4' \\ b_5' \\ b_6' \\ b_7' \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \times \begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \\ b_6 \\ b_7 \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{bmatrix}$$

$B_7\, b_6\, b_5\, b_4\, b_3\, b_2\, b_1\, b_0$ adalah urutan bit dalam elemen state atau *array byte* .

TABEL 2.2 Substitusi (S-Box)

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

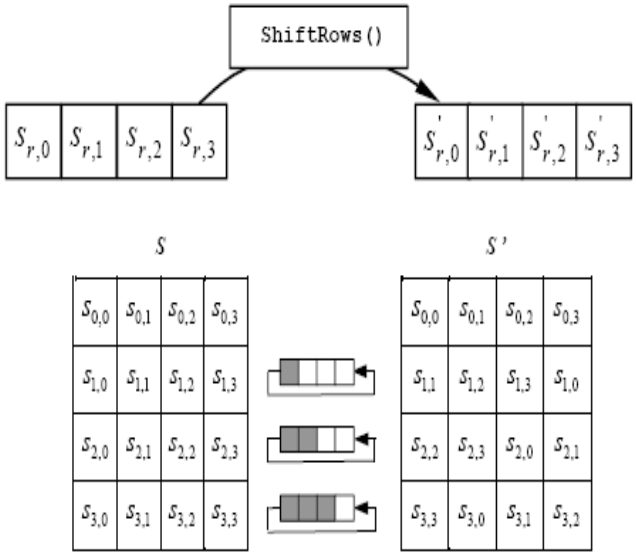
Sumber : (Vony Yuniati, 2009)



Sumber : (Vony Yuniati, 2009)
GAMBAR 2.4 SubBytes()

a. ShiftRows

Transformasi *Shiftrows* pada dasarnya adalah proses pergeseran bit dimana bit paling kiri akan dipindahkan menjadi bit paling kanan (rotasi bit). Transformasi ini diterapkan pada baris 2, baris 3, dan baris 4. Baris 2 akan mengalami pergeseran bit sebanyak satu kali, sedangkan baris 3 dan baris 4 masing – masing mengalami pergeseran bit sebanyak dua kali dan tiga kali.



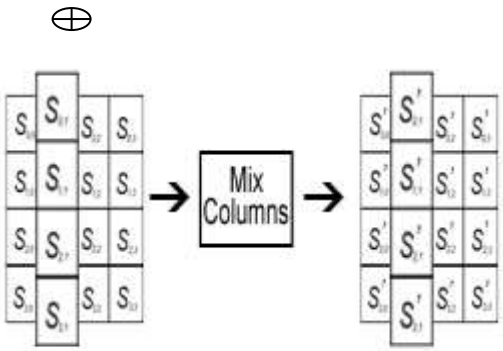
Sumber : (Vony Yuniati, 2009)
GAMBAR 2.5 Transformasi ShiftRows

b. MixColumns

Mixcolumns mengoperasikan setiap elemen yang berada dalam satu kolom. Elemen pada kolom dikalikan dengan suatu *polynomial* tetap a(x) = {03} x^3 + {01}x^2 + {01}x + {02}. Secara lebih jelas transformasi *mixcolumns* dapat dilihat pada perkalian matrik berikut ini :

$$\begin{bmatrix} s'_{0,c} \\ s'_{1,c} \\ s'_{2,c} \\ s'_{3,c} \end{bmatrix} = \begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \begin{bmatrix} s_{0,c} \\ s_{1,c} \\ s_{2,c} \\ s_{3,c} \end{bmatrix}$$

Melakukan proses penambahan pada operasi ini berarti melakukan operasi *bitwise* XOR ( ). Maka hasil dari perkalian matriks diatas dapat dianggap seperti perkalian yang ada di bawah ini :

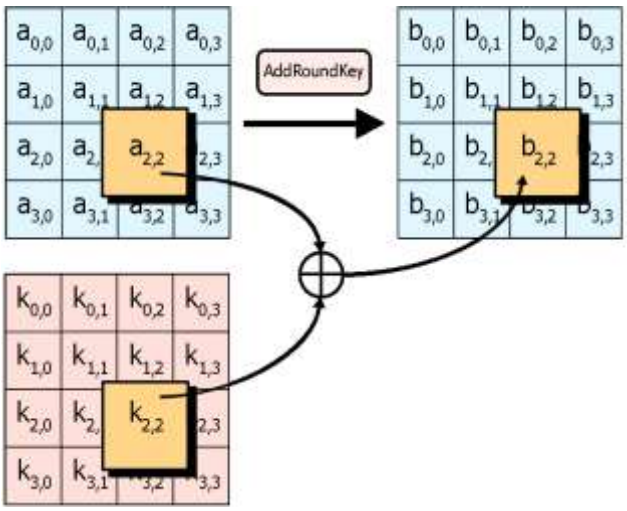


Sumber : (Stallings, 2003)
GAMBAR 2.6 MixColumns()

c. AddRoundkey

Pada proses *AddRoundKey*, sebuah *round key* ditambahkan pada *state* dengan operasi *bitwise* XOR. Setiap *roundkey* terdiri dari *Nb word* dimana tiap *word* tersebut akan dijumlahkan dengan *word* atau kolom yang bersesuaian dari *state* sehingga :

[w_i] adalah *word* dari *key* yang bersesuaian dimana i = *round* * Nb+c.Transformasi *AddroundKey* diimplementasikan pertama kali pada round = 0, dimana *key* yang digunakan adalah *initial key (key* yang dimasukkan oleh kriptografer dan belum mengalami proses *key expansion*).

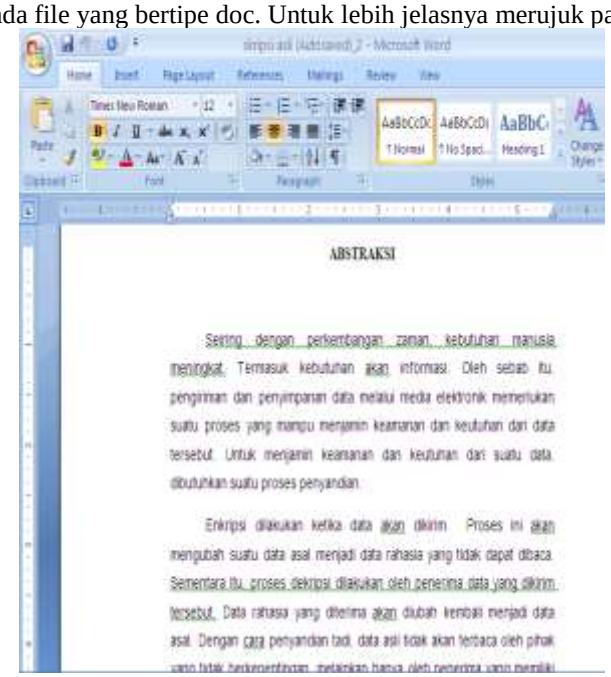


Sumber : (Surian, 2006)
GAMBAR 2.7 AddRoundKey

3. PEMBAHASAN

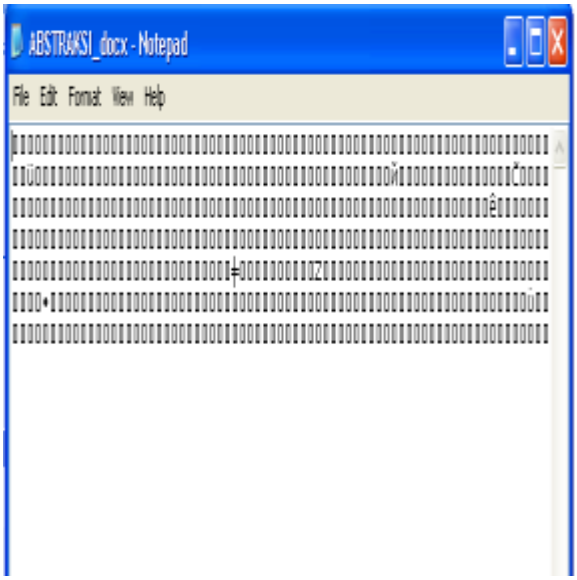
a. Berikut adalah hasil percobaan enkripsi dari beberapa file:

1. Enkripsi akan dilakukan pada file yang bertipe doc. Untuk lebih jelasnya merujuk pada Gambar 4.11.



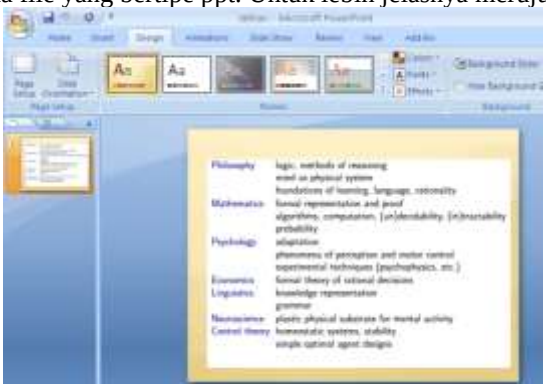
Gambar 4.11 File Enkripsi bertipe doc

2. Berikut adalah hasil enkripsi file bertipe doc yang dibuka dengan notepad. Untuk lebih jelasnya merujuk pada Gambar 4.12.



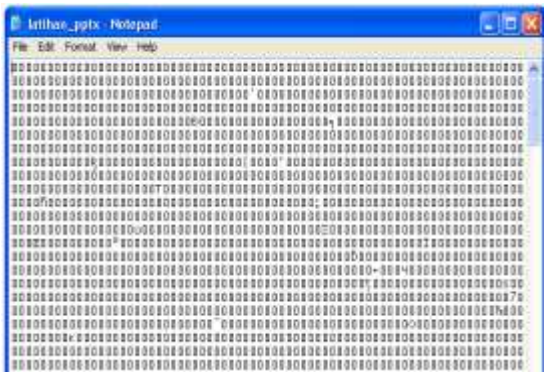
Gambar 4.12 Hasil Enkripsi file doc

3. Enkripsi akan dilakukan pada file yang bertipe ppt. Untuk lebih jelasnya merujuk pada Gambar 4.13.



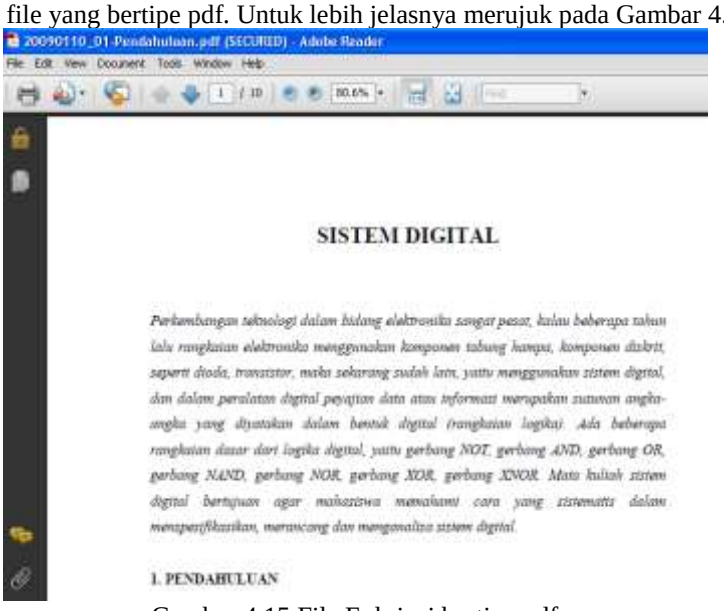
Gambar 4.13 File Enkripsi bertipe ppt

4. Berikut adalah hasil enkripsi file bertipe ppt yang dibuka dengan notepad. Untuk lebih jelasnya merujuk pada Gambar 4.14.



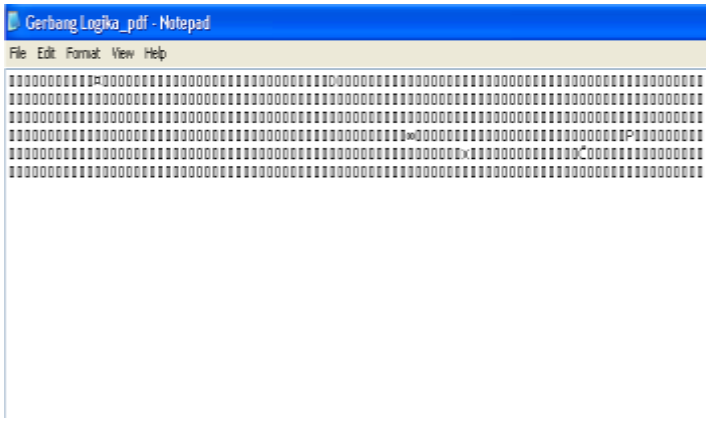
Gambar 4.14 Hasil Enkripsi file ppt

5. Enkripsi akan dilakukan pada file yang bertipe pdf. Untuk lebih jelasnya merujuk pada Gambar 4.15.



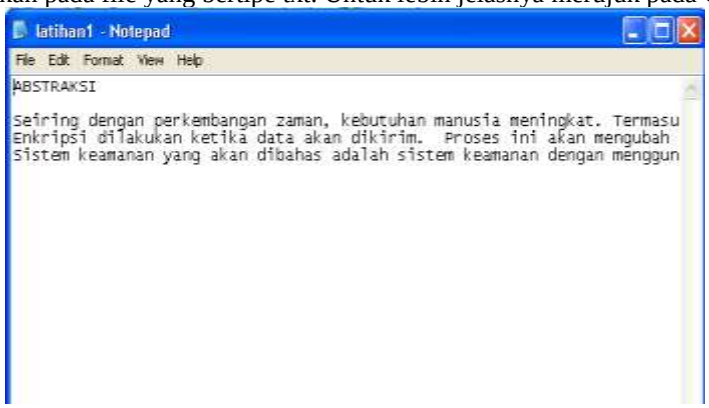
Gambar 4.15 File Enkripsi bertipe pdf

6. Berikut adalah hasil enkripsi file bertipe pdf yang dibuka dengan notepad. Untuk lebih jelasnya merujuk pada Gambar 4.16.



Gambar 4.16 Hasil Enkripsi file pdf

7. Enkripsi akan dilakukan pada file yang bertipe txt. Untuk lebih jelasnya merujuk pada Gambar 4.17.



Gambar 4.17 File Enkripsi Bertipe txt

8. Berikut adalah hasil enkripsi file bertipe pdf yang dibuka dengan notepad. Untuk lebih jelasnya merujuk pada Gambar 4.18.

Gambar 4.18 Hasil Enkripsi file txt

7 4. KESIMPULAN DAN SARAN

7.1 4.1. Kesimpulan

Dari hasil perancangan dan pembuatan program aplikasi kriptosistem dengan algoritma Rijndael ini dapat diambil kesimpulan sebagai berikut :

1. Program aplikasi ini dapat berjalan sesuai dengan teknis rancangan.
2. Program apilkasi ini akan membatasi orang yang tidak berhak terhadap isi pesan karena pesan telah di enkripsi.
3. Berdasarkan hasil perbandingan dengan aplikasi yang ada ternyata aplikasi yang dibuat masih membutuhkan waktu yang lebih lama untuk melakukan proses enkripsi dan dekripsi.

7.2 4.2. Saran

Saran – saran yang berguna untuk pengembangan program aplikasi ini adalah sebagai berikut :

1. Input untuk proses enkripsi tidak hanya dilakukan untuk data yang berformat teks saja, tetapi bisa untuk semua tipe data, juga bisa untuk data yang berupa suara,video dan lain sebagainya.
2. Format untuk penyimpanan data bisa hasil enkripsi diatur sesuai dengan keinginan pengguna.
3. Perlu adanya penyempurnaan dan perbaikan dari aplikasi yang dibuat agar *output* yang dihasikan lebih baik dari aplikasi yang sudah ada.

5. DAFTAR PUSTAKA

[1] Ariyus, D. (2005). *Kriptografi dan Keamanan Data Komunikasi.* Yogyakarta: Andi Offset.
D.Howard, J. (1997). *Security Incidents on the Internet.* USA: Sandia National Laboratories.
Gollmann, D. (2011). *Computer Security.* United States: John Wiley & Sons.

[2] Hartono. (2007). *Perancangan Aplikasi Kriptography Advanced Encryption Standard.*
Jogiyanto. (1990). *Analisis dan Disain Sistem Informasi.* Yogyakarta: Andi Offset.

[3] Kurniawan, Y. (2004). *Kriptografi Keamanan Internet dan Jaringan Komunikasi.* Bandung: Informatika.

[4] Nugroho, A. (2005). *Rational Rose untuk Pemodelan Berorientasi objek.* Bandung: Informatika.

[5] Saputra, E. H. (2005). *Kriptografi Dalam Aplikasi VB.NET*.

[6] Semuil Tjiharjadi, M. C. (2009). *Pengamanan Data Menggunakan Metoda Enkripsi Simetris dengan Algoritma Feal*. *Seminar Nasional Aplikasi Teknologi Informasi* .

[7] Nugroho, A. (2005). *Rational Rose untuk Pemodelan Berorientasi objek.* Bandung: Informatika.

[8] Stallings, W. (2003). *Cryptography and Network Security Principles and Practise.* New Jersey: Pearson Education.