Implementasi Enkripsi Pengiriman Gambar Sidik Jari Dengan Algoritma AES Untuk Autentifikasi User

Lingga Sartono, Bowo Nurhadiyono
Program Studi Teknik Informatika - S-1, Fakultas Ilmu Komputer
Universitas Dian Nuswantoro
Semarang, Indonesia

Abstract The web application is an application that is coded using the language supported web browser (such as HTML, JavaScript, AJAX, Java, etc.) that uses text base, it has a gap in terms of security. Where text or content can be copied and stolen easily. Therefore it is very important to design and build a program that maintained integrity, security, and also the confidentiality of such data, especially the data transmission which uses fingerprints for user authentication. So that the fingerprint image is unique and personal as user verification stay safe while. Therefore, the encryption of cryptographic techniques, which changed the original text message (plain text) into a text message that has been encrypted (cipher text), the data is secure. AES algorithm, is one of the cryptographic algorithms used to strengthen the security that has block size of 128 bits with technical substitution, permutation, and the number of turns per block. In this study, will be made an implementation of encryption data delivery with AES algorithm for user authentication. Fingerprint image used as an object for user authentication will be encrypted. From the results of the implementation of encryption sending pictures for user authentication produces the fingerprint image encryption.

Keywords : *Crypthograpic, AES.*

I. PENDAHULUAN

Aplikasi web adalah suatu aplikasi yang diakses menggunakan penjelajah web melalui suatu jaringan seperti Internet atau intranet. Ia juga merupakan suatu aplikasi perangkat lunak komputer yang dikodekan dalam bahasa yang didukung penjelajah web (seperti HTML, JavaScript, AJAX, Java, dll) [1].

Aplikasi web ternyata memiliki beberapa segi celah keamanan dan juga kekurangan dalam hal keamanannya. Yang mana text tersebut bisa disalin dan dicuri dengan mudah. Merupakan hal yang sangat penting bagi para perancang dan juga pembangun pemrogram untuk membuat aplikasi tersebut terjaga keintegritasannya, keamanannya, dan juga kerahasiaan datanya agar tidak dapat dicuri, disalin oleh pihak yang sangat tidak bertanggung jawab dalam pengoperasian aplikasi web tersebut.

Oleh sebab itu sangat penting untuk membuat solusi dalam bentuk nyata untuk pengamanan data tersebut terutama pada bagian pengiriman suatu data berupa gambar sidik jari tersebut yang digunakan sebagai suatu objek untuk authentifikasi tersebut Sehingga saat pengiriman data tersebut perlu dibuatnya suatu metode kriptografi, yaitu enkripsi untuk menjaga keamanan data tersebut.

Secara terminology kriptografi adalah ilmu dan seni untuk menjaga keamanan pesan ketika pesan dikirim dari suatu tempat ke tempat lain[2]. Yang menghasilkan suatu pesan teks asli (*plain text*) yang nantinya diubah menjadi suatu pesan teks yang sudah disandikan (*chipper text*) disebut enkripsi.

AES merupakan salah satu algoritma kriptografi simetris atau block chipper simetris untuk proses enkripsi yang memiliki ukuran panjang blok

dan kunci yang dapat dipilih secara independen 128, 192, dan 256 bit [3]. Algoritma AES (Rijndael) ini menggunakan beberapa teknik yang ada seperti substitusi, permutasi dan sejumlah putaran yang dikenakan pada tiap blok yang nantinya dilakukan untuk enkripsi dan dekripsi.

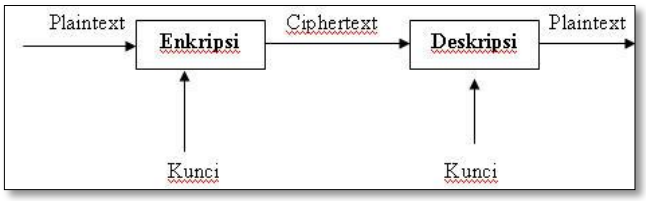
Oleh sebab itu, berdasarkan uraian yang ditulis, penulis bermaksud untuk mengambil tugas akhir (skripsi) dengan judul "Implementasi Enkripsi Pengiriman Gambar Sidik Jari dengan Algoritma AES untuk Authentifikasi User". Aplikasi web ini nantinnya dapat mengenkripsi *source code* pada gambar sidik jari yang dikirim ke server saat *user login*/authentifikasi user dengan algoritma tersebut.

II. METODE YANG DIUSULKAN

A. *Enkripsi*

Encryption adalah proses untuk mengubah *plaintext* ke *chipertext*.

Proses enkripsi dan deskripsi secara sederhana diterangkan sebagai berikut :



Gambar 1 Skema Enkripsi

EK (P) = C (Proses Enkripsi) **(2-1)**

DK (C) = P (Proses Dekripsi) **(2-2)**

Keterangan :

E : Enkripsi.

D : Deskripsi.

P : *Plain text* (Pesan sebelum dienkripsi).

C : *Cipher text* (Pesan setelah dienkrisi).

K : *Key* (Kunci).

Algoritma AES (Rijndael) ini menggunakan beberapa teknik yang ada seperti substitusi, permutasi dan sejumlah putaran yang dikenakan pada tiap blok yang nantinya dilakukan untuk enkripsi meliputi :

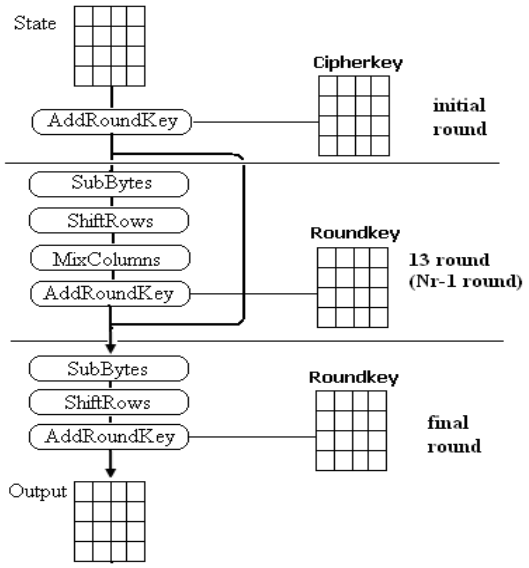
1. ***AddRoundKey,*** melakukan XOR antara *state* awal (plainteks) dengan *chiper key.*

2. Putaran sebanyak **Nr-1 kali**. Proses dilakukan pada putaran adalah:

   a. ***SubBytes*** adalah subtitusi byte dengan menggunakan tabel substitusi (S-Box).
   b. ***ShifRows*** adalah pergeseran baris – baris array secara *wrapping*.
   c. ***MixCoulumns*** adalah mengacak data di masing – masing kolom *array state*.
   d. ***AddRoundKey***

3. **Final round**, proses untuk putaran terakhir :

a. SubBytes
b. ShiftRows
c. AddRoundKey



Gambar 2 Diagram Proses Enkripsi

B. *Langkah Kerja*

1. Transformasi *SubBytes*

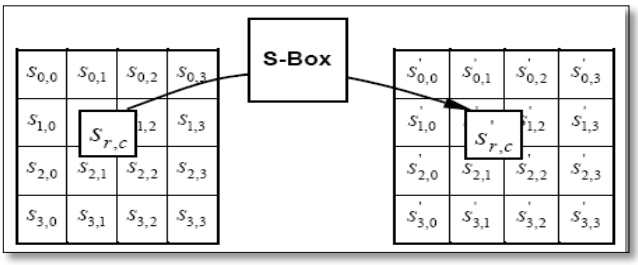
Langkah pada transformasi SubBytes adalah memetakan setiap byte dari *array state* menggunakan suatu tabel substitusi, S-Box.

hex	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

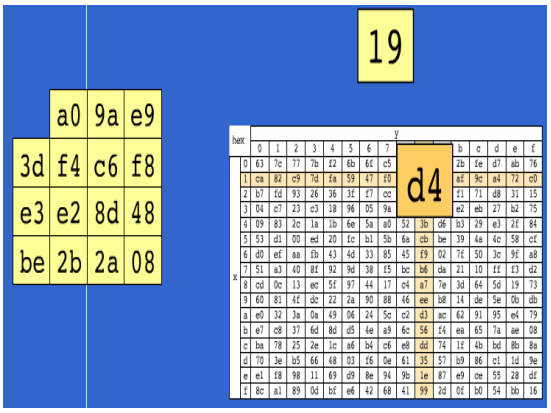
Tabel 1 Tabel S-Box Dikutip dari: Studi Algoritma Rijndael dalam Sistem Keamanan Data, 2009

Untuk pensubstitusiannya, jika setiap byte pada array state S[r,c]=xy, xy adalah digit heksadesimal dari nilai

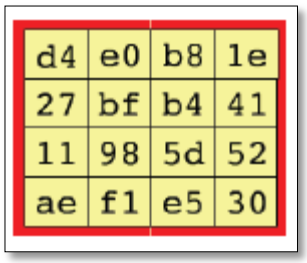
S[r,c], maka nilai substitusi dinyatakan dengan S'[r,c], merupakan elemen di dalam S-Box sebagai perpotongan baris x dan kolom y.



Gambar 3 Transformasi dari SubBytes Dikutip dari: Studi Algoritma Rijndael dalam Sistem Keamanan Data, 2009



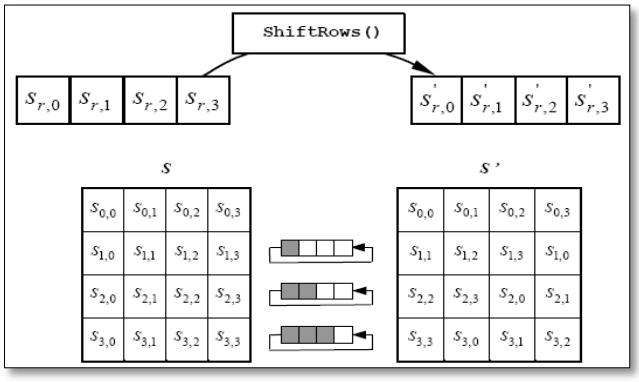
Gambar 4 Proses dari SubBytes Dikutip dari: Studi Algoritma Rijndael dalam Sistem Keamanan Data, 2009



Gambar 5 Hasil dari SubBytes Dikutip dari: Studi Algoritma Rijndael dalam Sistem Keamanan Data, 2009

2. ***ShiftRows***

Transformasi ShiftRows adalah proses pergeseran secara wrapping pada 3 baris terakhir dan bit pada bit paling kiri akan dipindah menjadi bit paling kanan sesuai jumlah pergeseran. Pergeseran ini hanya diterapkan pada baris r=1, r=2, dan r=4. Baris r=1 akan mengalami pergeseran bit sebanyak satu kali, sedangkan baris r=3 dan r=4 masing-masing mengalami pergeseran bit sebanyak dua kali dan tiga kali.



Gambar 6 Transformasi ShiftRows Dikutip dari: Studi Algoritma Rijndael dalam Sistem Keamanan Data, 2009

3. ***MixColumns***

Mixolumns mengalikan setiap elemen yang berada dalam satu kolom dari *array state* dengan polinom a(x)mod(x^4+1). Setiap kolomnya diperlakukan sebagai polinom 4 suku pada GF(2^8) :

Polinom a(x) :

$$a(x) = \{03\}x_3 + \{01\}x_2 + \{01\}x + \{02\}$$

Transformasi dinyatakan dalam perkalian matrix :

$$s'(x) = a(x) \otimes s(x)$$

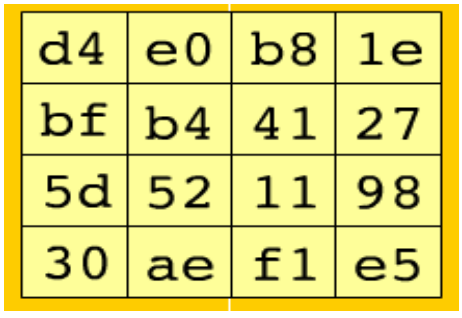
$$\begin{bmatrix} s'_{0,c} \\ s'_{1,c} \\ s'_{2,c} \\ s'_{3,c} \end{bmatrix} = \begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \begin{bmatrix} s_{0,c} \\ s_{1,c} \\ s_{2,c} \\ s_{3,c} \end{bmatrix}$$

$$s'_{0,c} = (\{02\} \bullet s_{o,c}) \oplus (\{03\} \bullet s_{1,c}) \oplus s_{2,c} \oplus s_{3,c}$$

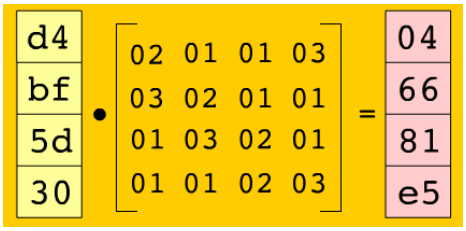
$$s'_{1,c} = s_{0,c} \oplus (\{02\} \bullet s_{1,c}) \oplus (\{03\} \bullet s_{2,c}) \oplus s_{3,c}$$

$$s'_{2,c} = s_{0,c} \oplus s_{1,c} \oplus (\{02\} \bullet s_{2,c}) \oplus (\{03\} \bullet s_{3,c})$$

$$s'_{3,c} = (\{03\} \bullet s_{0,c}) \oplus s_{1,c} \oplus s_{2,c} \oplus (\{02\} \bullet s_{3,c})$$



Gambar 7 Hasil ShifRows Dikutip dari: Studi Algoritma Rijndael dalam Sistem Keamanan Data, 2009



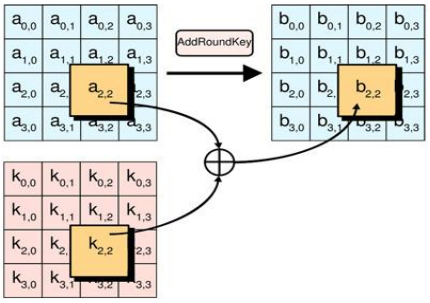
Gambar 9 Operasi MixColumns Dikutip dari: Studi Algoritma Rijndael dalam Sistem Keamanan Data, 2009

04	e0	48	28
66	cb	f8	06
81	19	d3	26
e5	9a	7a	4c

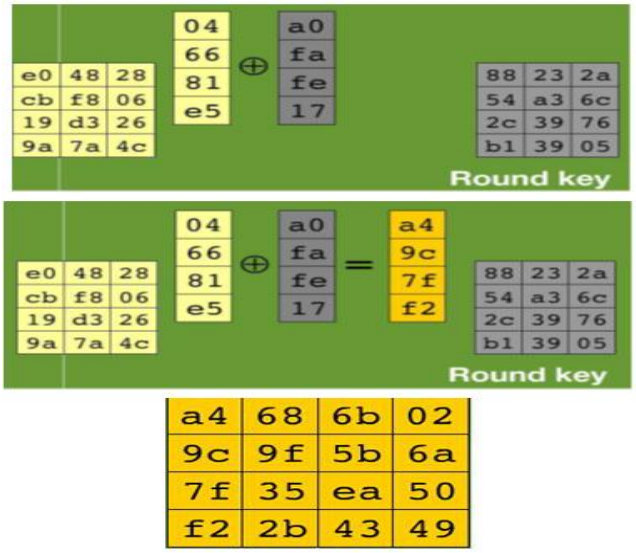
Gambar 10 Hasil keseluruhanMixColumns Dikutip dari: Studi Algoritma Rijndael dalam Sistem Keamanan Data, 2009

4. *AddRoundKey*

Transformasi ini melakukan XOR pada sebuah *round key* dengan *array state*, serta hasilnya disimpan di *array state*.



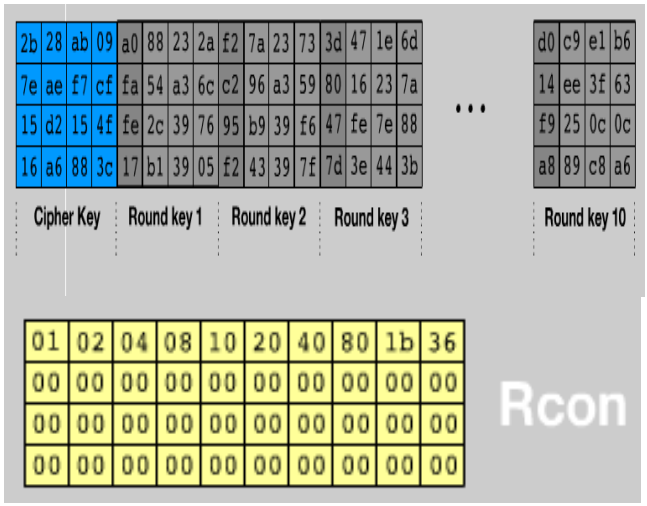
Gambar 11 Transformasi AddRoundKey Dikutip dari: Studi Algoritma Rijndael dalam Sistem Keamanan Data, 2009



Gambar 12 Hasil AddRoundKey Dikutip dari: Studi Algoritma Rijndael dalam Sistem Keamanan Data, 2009

5. Ekspansi kunci

Algoritma Rijndael melaksanakan cipher key dan membuat suatu ekspansi kunci untuk menghasilkan suatu key schedule. Jika ekspansi kunci yang diperlukan Rijndael Nb(Nr+1) word, sehingga bisa digunakan AES 128 bit, maka 4(10+1)=40 word=44x32 bit=1408 bit subkey. Ekspansi dari 128 menjadi 1408 bit subkey, proses ini disebut dengan key schedule. Subkey ini diperlukan karena setiap round merupakan suatu inisial dari Nb word untuk Nr=0 dan Nb untuk Nr=1,3 untuk Nr= 2,....,11 Nb untuk Nr=10, dari operasi ini akan didapatkan schedule kunci yang berisi array linier 4 byte word (wi), 0=i(Nr+1).



Gambar 13 Proses ekspansi kunci Dikutip dari: Studi Algoritma Rijndael dalam Sistem Keamanan Data, 2009

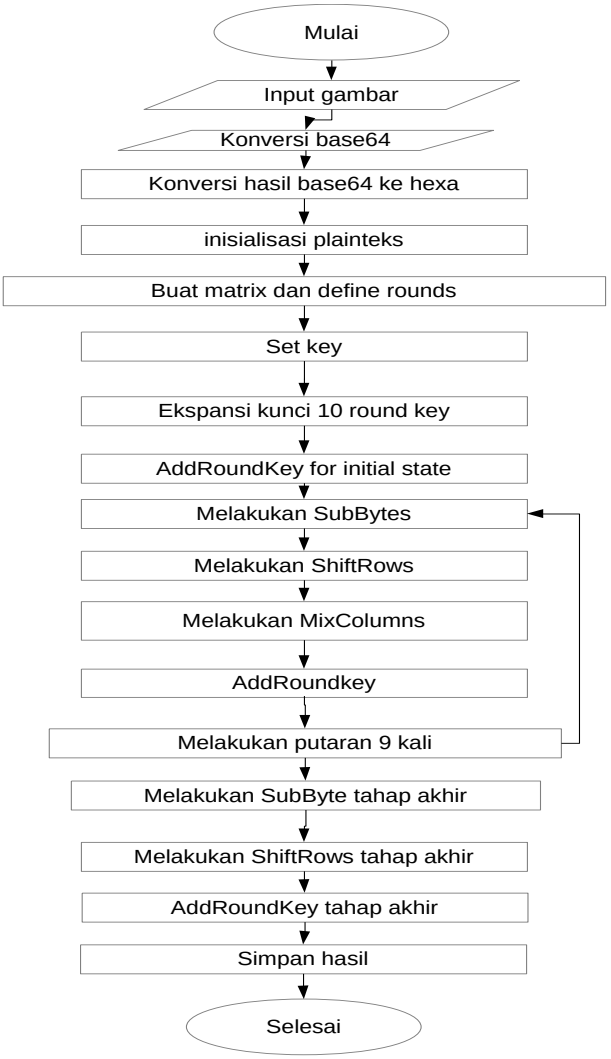
III. IMPLEMENTASI

Pada bagian ini menjelaskan langkah proses enkripsi.

1. Input, objek inputan gambar.
2. Konversi gambar ke base64.
3. Konversikan teks tersebut ke dalam bentuk bit ke hexadesimal.
4. Inisialisasi plainteks.
5. Setelah itu buat matrix dan define rounds.
6. Siapkan kunci array 4x4.
7. Ekpansi kunci , sejumlah 10 round key pada kunci yang digunakan.
8. Lakukan AddRoundkey, tiap kolom sebagai intial state.
9. Hasil dari addRoundKey dilakukan SubBytes, substitusi bytenya dengan tabel S-box.
10. Melakukan ShiftRows dari hasil SubBytes, geser 3 baris array terakhir berurutan geser 1,2,3 bit.
11. Melakukan MixColoumns, dengan tabel Galois Field .

12. Melakukan AddRoundkey.
13. Melakukan putaran sebanyak 9 kali pada langkah 7-8 atau sampai sebelum putaran terakhir.
14. Lakukan SubBytes tahap terakhir setelah putaran ke 9.
15. Lakukan ShiftRows tahap terakhir.
16. AddRoundkey terakhir

Untuk proses detailnya dapat dilihat pada gambar berikut :

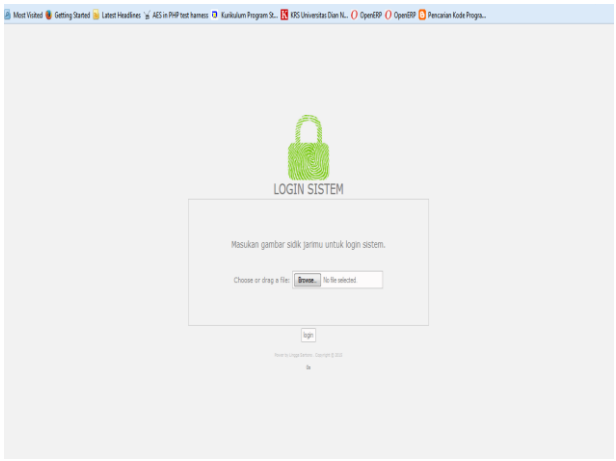


Gambar 14 Proses Enkripsi

IV. HASIL & PEMBAHASAN

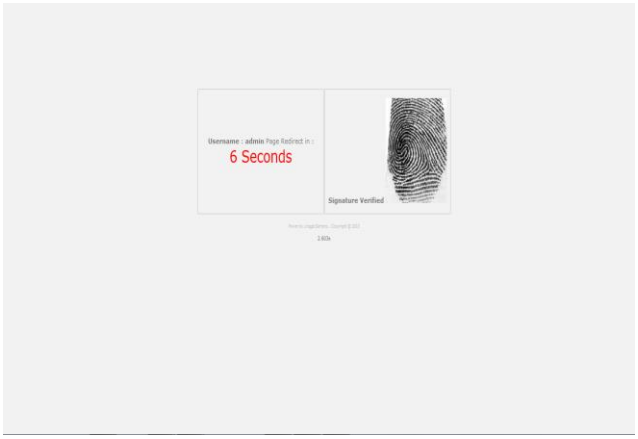
A. *Penerapan*

Implementasi enkripsi pengiriman gambar dengan algoritma AES untuk authentifikasi user diterapkan pada sistem informasi perpustakaan online dan dapat dengan browser manapun, pada bagian ini Penulis menggunakan fire fox untuk menampilkan hasil implementasi:



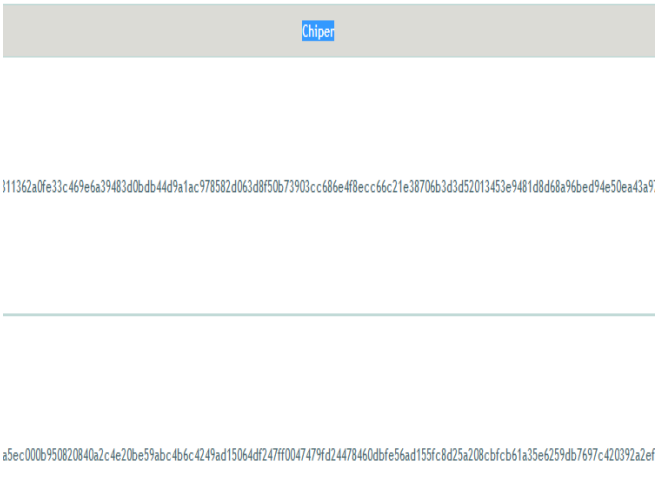
Gambar 15 Tampilan Login Sistem

Pada gambar tampilan login sistem ini dilakukan pemilihan gambar yang digunakan untuk autentifikasi user.



Gambar 16 Tampilan Verifikasi

Pada tampilan verifikasi inilah proses enkripsi sudah terjadi, sehingga menghasilkan *chipper* yang digunakan untuk verifikasi user yang selanjutnya menuju pada Menu sistem perpustakaan online.



Gambar 18 Hasil enkripsi

B. Pengujian

Tabel 2 Pengujian

Iterasi ke-	Enter Sis	Gambar =benar	Same key	Enkrip	Dekrip ke Gambar	Masuk Halaman Menu	Keterangan
1	O	-	-	-	-	-	*Berhasil ke halaman utama*
2	O	O	O	O	-	-	*Berhasil merubah Gambar menjadi Cipherteks*
3	O	O	O	O	O	-	Berhasil merubah *Cipherteks* menjadi *Plainteks ke Gambar*
4	O	O	O	O	O	O	Berhasil masuk sistem tanpa ada kesalahan
5	O	X	-	-	-	-	Gagal login, gambar salah
6	O	O	X	O	X	O	Gagal dekrip ke gambar karena kunci di ubah, dengan gambar sama.
7	O	O	X	O	X	O	Gagal dekrip ke gambar ketika kunci berbeda lagi dengan yang sebelumnya.
8	O	O	O	O	X	O	Gagal dekripsi ke gambar, dengan kunci sama pada tahap sebelumnya.
9	O	O	O	O	O	O	Berhasil ketika gambar dan kunci selalu sama.

Hasil pengujian oleh dapat didokumentasikan dalam bentuk tabel pengujian yang disertai keterangannya. Tabel ini digunakan untuk mengukur tingkat keberhasilan aplikasi pada tiap tahapan yang dilalui serta bahan untuk perencanaan iterasi berikutnya.

Keterangan kode yang digunakan:
O = Ya
X = Tidak
- = Tidak dilakukan pengujian pada variabel pengujian

Dari pengujian pengujian tersebut dapat disimpulkan bahwa aplikasi telah memenuhi tujuan penelitian ini. Enkripsi Pengiriman Gambar Sidik Jari berjalan dengan baik dan\ ditampilkan dengan baik sesuai harapan.

V. PENUTUP

Dari analisis, perancangan dan implementasi fungsi algoritma AES pada pengiriman gambar sidik jari untuk autentifikasi user, dapat disimpulkan bahwa :

1. Fungsi Algoritma AES dapat diterapkan dengan baik pada aplikasi web sistem informasi perpustakaan online untuk authentifikasi user.
2. Enkripsi dan dekripsi gambar juga berjalan dengan baik, begitupun dengan pengiriman gambar untuk autentifikasi user.
3. Perubahan key,data tetap aman dan gambar tidak tampil ketika di dekripsi.
4. Perubahan *cipher text*, gambar juga tidak dapat tampil ketika proses dekripsi.
5. Terlihat tingkat perbedaan keamanan saat pengiriman gambar, sudah terenkripsi.

REFERENCES

[1] http://id.wikipedia.org/wiki/Aplikasi_web (diakses pada 18 February 2015)

[2] Ariyus, D. (2008). Pengantar ilmu kriptografi: teori, analisis, dan implementasi. Yogyakarta: Andi.

[3] William, S., & Stallings, W.. (2006) *Cryptography and Network Security, 4/E*. Pearson Education India.

[4] Satria, Eko. (2009) "Studi Algoritma Rijndael dalam Sistem Keamanan Data.".

[5] Widiasari, Indrastanti R. 2012 "Combining Advanced Encryption Standard (AES) and One Time Pad (OTP) Encryption for Data Security." *International Journal of Computer Applications* 57.20.

[6] Rosyadi, Ahmad. (2012) "Implementasi Algoritma Kriptografi AES Untuk Enkripsi Dan Dekripsi Email." *TRANSIENT* 1.3

[7] Yuniati, Voni, Gani Indriyanta, and Antonius Rachmat C. (2011) "Enkripsi dan Dekripsi dengan Algoritma Aes 256 Untuk Semua Jenis File." *Jurnal Informatika* 5.1.

[8] Surian, Didi. (2009) "Algoritma Kriptografi AES Rijndael." *TESLA Jurnal Teknik Elektro UNTAR* 8.2: pp-97.

[9] Di, K. K. D. P. T., & Kelas, V. I. I. (2000). Metodologi penelitian.

[10] Fahmi, H., & Faidah, H. (2010). Aplikasi Kritpografi Modern untuk Pengiriman Data Teramankan. MH Thamrin, 8.

[11] Wang, P. T., & Wu, S. M. (2001). *U.S. Patent Application 09/849,279*.

[12] Wong, J. Y. (2006). *U.S. Patent No. 7,039,223*. Washington, DC: U.S. Patent and Trademark Office.

[13] http://en.wikipedia.org/wiki/Base64 (diakses 25 February 2015)

[14] http://id.wikipedia.org/wiki/Diagram_alir (diakses pada 22 February 2015)