KRIPTOGRAFI CITRA DIGITAL DENGAN ALGORITMA RIJNDAEL DAN TRANSFORMASI WAVELET DISKRIT HAAR

Bagus Satrio Waluyo Poetro, Aris Sugiharto dan Sukmawati Nur Endah

Program Studi Teknik Informatika Universitas Diponegoro
Jl. Prof Soedharto, Kampus UNDIP Tembalang Semarang
bagusswp@yahoo.com, aris.sugiharto@undip.ac.id, sukma_ne@undip.ac.id

Abstrak

Citra merupakan salah satu bentuk data atau informasi penting saat ini. Dikarenakan sebuah kepentingan dipandang perlu untuk mengamankan informasi yang ada pada sebuah citra. Salah satunya adalah dengan menggunakan kriptografi. Pada penelitian ini digunakan algoritma Rijndael dan transformasi wavelet diskrit untuk proses kriptografi sekaligus mempercepat prosesnya. Pada algoritma Rijndael digunakan kunci sebesar 128, 192 maupun 256 bit. Dari hasil pengujian dengan menggun akan citra digital berukuran berbeda, baik RGB maupun grayscale menghasilkan tingkat keberhasilan > 90%.

Kata kunci : kriptografi, Rijndael, transformasi wavelet diskritr

1. Pendahuluan

Pada era modern saat ini banyak sekali kemungkinan penyadapan data, maka aspek keamanan dalam pertukaran informasi menjadi sangat penting karena suatu komunikasi data jarak jauh belum tentu memiliki jalur transmisi yang aman dari penyadapan sehingga keamanan informasi menjadi bagian penting dalam dunia informasi itu sendiri.

Citra merupakan salah satu sumber informasi, ini dikarenakan citra kaya akan informasi yang dibutuhkan. Citra yang berukuran besar menimbulkan masalah pada penyimpanan, pengenkripsian dan pengiriman citra, yaitu kebutuhan media penyimpanan data yang besar serta pengenkripsian dan waktu pengiriman yang lama[1].

Kriptografi merupakan salah satu teknik yang digunakan untuk meningkatkan aspek keamanan suatu informasi. Algoritma kriptografi yang baik akan memerlukan waktu yang lama untuk memecahkan data yang telah disandikan.

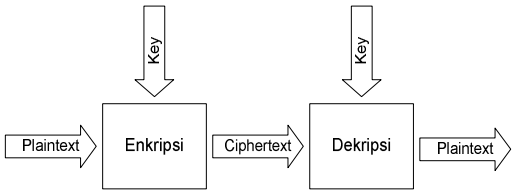
Transformasi wavelet diskrit merupakan salah satu teknik pemrosesan sinyal digital yang lebih mudah diaplikasikan dan hasilnya lebih bagus dibandingkan transformasi Fourier. Transformasi wavelet diskrit secara umum merupakan dekomposisi citra pada frekuensi subband citra itu sendiri[2].

Dengan melihat konsep tersebut, dalam paper ini akan digunakan kriptografi dengan algoritma Rijndael dan transformasi wavelet diskrit Haar untuk menangani keamanan dalam penyimpanan citra digital yang bersifat rahasia.

2. Tinjauan Pustaka

Secara harfiah, *image* atau citra merupakan gambar pada bidang dwimatra (dua dimensi). Sedangkan dilihat dari sudut pandang matematis, citra merupakan fungsi kontinu atau menerus dari intensitas cahaya pada bidang dwimatra[3]. Citra terdiri dari dua macam jenis, yaitu citra kontinu dan citra diskrit. Citra kontinu berasal dari sistem optik yang menerima sinyal analog. Sedangkan citra diskrit berasal dari proses digitalisasi terhadap citra kontinu. Representasi citra dari fungsi kontinu menjadi nilai-nilai diskrit disebut sebagai digitalisasi. Citra yang dihasilkan inilah disebut citra digital.

Kriptografi adalah ilmu yang mempelajari tentang cara menjaga keamanan suatu pesan atau informasi. Pesan atau informasi dapat dikategorikan ke dalam dua jenis, yaitu pesan yang dapat dibaca dengan mudah (plaintext) dan pesan yang tidak mudah dibaca (ciphertext). Algoritma kriptografi dibagi menjadi dua bagian yaitu enkripsi dan dekripsi. Enkripsi adalah proses mengubah plaintext menjadi ciphertext, sedangkan dekripsi adalah kebalikannya yaitu mengubah ciphertext menjadi plaintext. Proses kriptografi dapat dilihat pada gambar 1.



Gambar 1. Proses kriptografi

Algoritma Rijndael adalah pemenang sayembara terbuka yang diadakan oleh NIST (*National Institute of Standards and Technology*) pada tahun 1999 bersamaan dengan 4 finalis lainnya yaitu algoritma Serpent, MARS, Twofish, dan RC6 yang bertujuan untuk membuat standard algoritma kriptografi yang baru sebagai pengganti Data Encryption Standard (DES). Algoritma Rijndael meliputi tiga tipe kunci yaitu kunci berkapasitas 256 bit, 192 bit, dan 128 bit. Besar kapasitas kunci berpengaruh terhadap jumlah putaran (*round*) yang diimplementasikan dalam algoritma ini[4].

Tabel 1. Perbandingan jumlah blok kunci, panjang kunci per blok dan jumlah putaran pada tiga tipe algoritma Rijndael

	Jumlah Blok Kunci	Panjang Kunci per Blok	Jumlah Putaran
Rijndael – 128	4	4	10
Rijndael – 192	6	4	12
Rijndael – 256	8	4	14

Blok kunci dalam algoritma Rijndael ini adalah karakter atau huruf yang dikelompokkan sebagai kunci, sedangkan panjang kunci merupakan banyaknya karakter atau huruf per blok. Jumlah putaran dalam algoritma ini menyatakan banyaknya putaran atau looping dalam memproses data masukan.

Bentuk kunci di dalam algoritma Rijndael bermacam – macam tergantung implementasinya. Kunci bisa berupa teks yang dipecah per blok dan ada juga yang berupa matriks. Kunci berupa teks biasa diimplementasikan terhadap sebuah kumpulan teks seperti sms (*short messaging service*), *e-mail*, maupun file dari program pengolah kata seperti microsoft word. Kunci berupa matriks digunakan pada data berupa citra (*image*).

Proses yang dilakukan terdiri dari dua langkah yaitu proses enkripsi dan proses dekripsi. Di dalam algoritma Rijndael, semua proses dilakukan dalam bentuk heksadesimal. Proses perubahan dalam notasi biner, desimal, dan heksadesimal dalam algoritma Rijndael mengacu pada tabel ASCII.

Proses enkripsi pada algoritma Rijndael terdiri dari 4 jenis transformasi bytes, yaitu SubBytes, ShiftRows, Mixcolumns, dan AddRoundKey. Sama halnya dengan proses enkripsi, proses dekripsi juga terdiri dari 4 jenis transformasi bytes yaitu Inv SubBytes, InvShiftRows, InvMixcolumns, dan AddRoundKey.

Proses SubBytes merupakan transformasi byte dengan setiap elemen pada data masukan (*state*) akan dipetakan dengan menggunakan sebuah tabel substitusi (S-Box). Sedangkan untuk *InvSubBytes* hanya berbeda tabel substitusinya (Inv 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

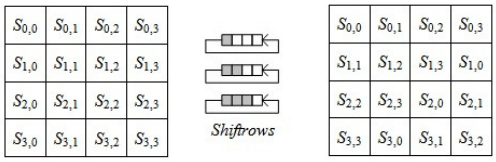
Gambar 2. Tabel substitusi S-Box

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

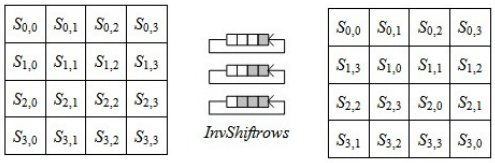
Gambar 3. Tabel substitusi Inv S-Box

Transformasi *Shiftrows* pada dasarnya adalah proses pergeseran *byte* dengan *byte* paling kiri akan dipindahkan menjadi *byte* paling kanan (rotasi). 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. Sedangkan transformasi *InvShiftrows* merupakan proses pergeseran *byte*

dengan arah yang berlawanan dengan transformasi *Shiftrows* yaitu ke arah kanan.



Gambar 4. Proses Shiftrows



Gambar 5. Proses InvShiftrows

Proses *Mixcolumns* mengoperasikan setiap elemen yang berada dalam satu kolom pada state. Elemen pada kolom dikalikan dengan suatu polinomial tetap. Proses *InvMixcolumns* sama seperti transformasi *Mixcolumns* hanya berbeda polinomial pengalinya.

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

Gambar 6. Polinomial tetap pada *Mixcolumns*

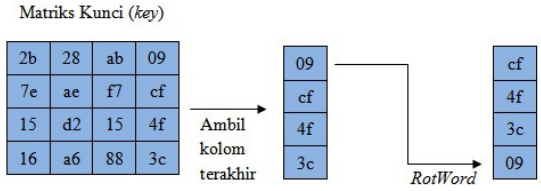
$$\begin{bmatrix} s'_{0,c} \\ s'_{1,c} \\ s'_{2,c} \\ s'_{3,c} \end{bmatrix} = \begin{bmatrix} 0e & 0b & 0d & 09 \\ 09 & 0e & 0b & 0d \\ 0d & 09 & 0e & 0b \\ 0b & 0d & 09 & 0e \end{bmatrix} \begin{bmatrix} s_{0,c} \\ s_{1,c} \\ s_{2,c} \\ s_{3,c} \end{bmatrix}$$

Gambar 7. Polinomial tetap pada *InvMixcolumns*

Proses transformasi terakhir adalah *AddRoundKey* yang merupakan penjumlahan bitwise XOR antara plaintext dengan *key* (dirubah ke dalam notasi biner terlebih dahulu).

Selain proses enkripsi dan dekripsi, algoritma Rijndael juga memiliki kemampuan untuk mengekspansi kunci (*key expansion*) yang digunakan pada setiap putaran dalam proses enkripsi maupun dekripsi. Ekspansi kunci dalam setiap putaran disebut juga *key schedule*. Proses *key schedule* meliputi beberapa fungsi yaitu fungsi *RotWord* , *SubBytes*, dan *Rcon*.

Fungsi *RotWord* mengambil satu kolom terakhir dari matriks kunci utama yang telah dimasukan oleh pengguna kemudian melakukan satu permutasi siklik yaitu *byte* paling atas dirotasi menjadi *byte* paling bawah.



Gambar 8. Contoh fungsi *RotWord*

Fungsi *SubBytes* dalam *key schedule* sama seperti transformasi *SubBytes* pada proses enkripsi. Fungsi *Rcon* merupakan matriks ketetapan yang berukuran 10x4 dan digunakan untuk operasi XOR dalam proses *key schedule*.

01	02	04	08	10	20	40	80	1b	36
00	00	00	00	00	00	00	00	00	00
00	00	00	00	00	00	00	00	00	00
00	00	00	00	00	00	00	00	00	00

Gambar 9. *Rcon*

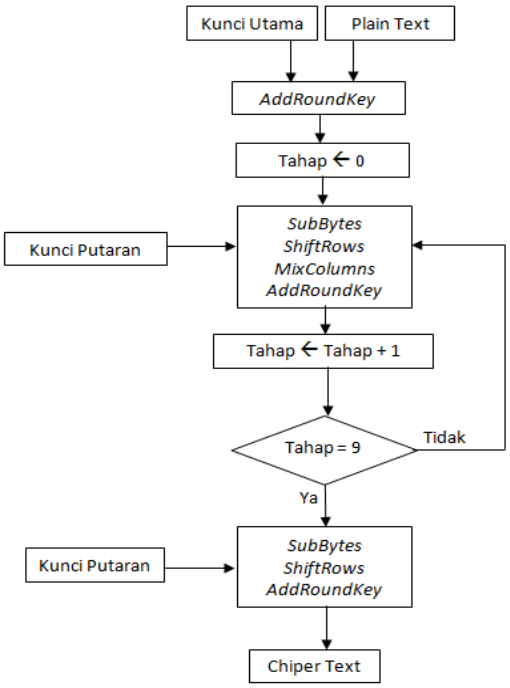
Tahap – tahap penyandian dalam algoritma Rijndael untuk proses enkripsi dan dekripsi secara urut dapat dijelaskan sebagai berikut :

Proses Enkripsi

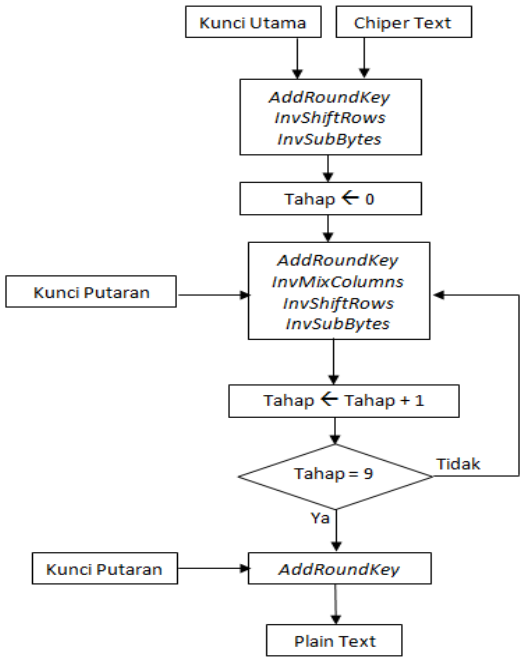
1. MengXORkan plaintext dengan kunci masukan (*AddRoundKey*).
2. Mensubtitusi plaintext (*SubBytes*).
3. Melakukan proses pergeseran bit terhadap plaintext (*Shiftrows*).
4. Mengalikan kolom – kolom plaintext dengan kolom – kolom matriks polinomial yang sudah ditentukan (*Mixcolumns*).
5. Melakukan proses *AddRoundKey* lagi tetapi dengan kunci hasil *Key Schedule* setiap putaran.
6. Mengulangi (*looping*) langkah dua sampai langkah lima sebanyak sembilan kali.
7. Mensubtitusi plaintext (SubBytes).
8. Melakukan proses pergeseran bit (Shiftrows).
9. Melakukan proses AddR*oundKey* dengan kunci hasil *Key Schedule* putaran kesepuluh dan menghasilkan ciphertext.

Proses Dekripsi

1. MengXORkan ciphertext dengan kunci masukan (*AddRoundKey*).
2. Melakukan proses pergeseran bit terhadap ciphertext (*InvShiftrows*).
3. Mensubtitusi ciphertext (*InvSubBytes*).
4. Melakukan proses *AddRoundKey* lagi tetapi dengan kunci hasil *Key Schedule* setiap putaran.
5. Mengalikan kolom – kolom ciphertext dengan kolom – kolom matriks polinomial yang sudah ditentukan (*InvMixcolumns*).
6. Melakukan proses pergeseran bit terhadap ciphertext (*InvShiftrows*).
7. Mensubtitusi ciphertext (*InvSubBytes*).
8. Mengulangi (*looping*) langkah empat sampai langkah tujuh sebanyak sembilan kali.
9. Melakukan proses *AddRoundKey* dengan kunci hasil *Key Schedule* putaran kesepuluh dan menghasilkan plaintext.

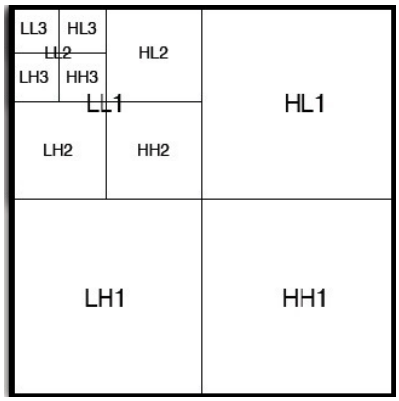


Gambar 10. Diagram proses enkripsi



Gambar 11. Diagram proses dekripsi

Transformasi wavelet diskrit secara umum merupakan dekomposisi citra pada frekuensi subband citra tersebut. Komponen subband transformasi wavelet dihasilkan dengan cara penurunan level dekomposisi. Implementasi transformasi wavelet diskrit dapat dilakukan dengan cara melewatkan sinyal melalui sebuah tapis lolos rendah (*low pass filter*/LPF) dan tapis lolos tinggi (*high pass filter*/HPF) dan melakukan downsampling pada keluaran masing masing filter.
Wavelet diskrit metode Haar mendekomposisikan dengan nilai *low pass filter* dan *high pass filter*[5].



Gambar 12. Transformasi wavelet diskrit Haar

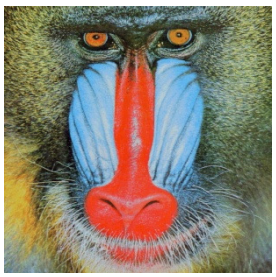
3. Metode PenelitiaN

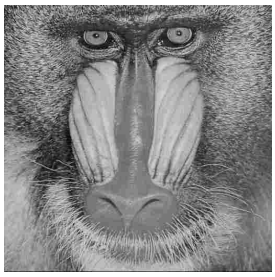
Metode yang digunakan adalah percobaan pada tiga citra digital berukuran berbeda, empat ekstensi file dan terdiri dari dua komposisi warna yaitu RGB dan *grayscale*. Citra digital yang digunakan adalah citra Lena, Mandrill dan Peppers dengan ukuran 256x256, 512x512, 1024x1024 piksel berekstensi .jpg, .bmp, .png, dan .tif. Percobaan dilakukan untuk kedua proses yaitu proses enkripsi dan proses dekripsi.

4. Hasil Dan Pembahasan

Hasil proses enkripsi dan dekripsi pada citra digital Lena, Mandrill dan Peppers dapat dilihat pada tabel 2 berikut:

Tabel 2. Hasil percobaan enkripsi dan dekripsi pada citra RGB maupun Grayscale

Komposisi Warna	Citra Awal / Hasil Dekripsi	Citra Hasil Enkripsi		
		Level 4	Level 5	Level 6
RGB				
				----
			----	----

Grayscale	 M			
	 F			----
			----	----

.Dari hasil di atas dapat dilihat bahwa dari beberapa gambar yang berekstensi berbeda dan ukuran bervariasi akan menampilkan citra hasil yang berukuran sama dan memiliki tampilan sama persis dalam setiap level transformasinya. Dalam tabel 2 ada kolom yang tidak terisi, hal ini dikarenakan ukuran citra hasil hanya terbatas sampai ukuran tertentu.

5. KESIMPULAN

Berdasarkan hasil penelitian yang telah dilaporkan di dalam paper ini, maka dapat diambil kesimpulan sebagai berikut:

- Kriptografi citra digital menggunakan algoritma Rijndael dan transformasi wavelet diskrit Haar dapat menghasilkan suatu citra digital yang tidak jelas objeknya pada saat proses enkripsi dan menghasilkan suatu citra digital yang jelas objeknya saat proses dekripsi.
- Proses kriptografi dapat dilakukan pada citra RGB maupun citra *grayscale*.
- Semakin besar level transformasi citra semakin cepat juga proses kriptografi yang dilakukan karena ukuran citra semakin kecil.
- Setiap citra yang terenkripsi akan kembali seperti semula saat proses dekripsi.

Referensi

[1] S.W.P Bagus , *"Kriptografi File Citra Digital Menggunakan Algoritma Rijndael dan Transformasi Wavelet Diskrit"*, Semarang , Program Studi Teknik Informatika Universitas Diponegoro, 2010.

[2] Fajri, 2006, *fajri.freebsd.or.id/tugas_akhir/bab2.pdf*, diakses tanggal 8 Desember 2009.

[3] Munir, R., 2004, *"Pengolahan Citra Digital dengan Pendekatan Algoritmik",* Bandung : Penerbit Informatika.

[4] Wibowo, W.A., 2004, *"Advanced Encryption Standard, Algoritma Rijndael",* Bandung : Institut Teknologi Bandung.

[5] Medison, A.S, 2007, "*Sistem Pembanding Citra Pas Foto dengan Metode Transformasi Wavelet*", Sumatera Utara : Universitas Sumatera Utara