ANALISIS KRIPTOGRAFI SIMETRIS AES DAN KRIPTOGRAFI ASIMETRIS RSA PADA ENKRIPSI CITRA DIGITAL

[1]Geby Geta Putri, [2]Wiwin Styorini, [3]Rizki Dian Rahayani

[1,2,3]Program Studi Teknik Elektro Telekomunikasi, Politeknik Caltex Riau,
email: [1]geby13tet@mahasiswa.pcr.ac.id, [2]wiwin@pcr.ac.id, [3]uki@pcr.ac.id,

Abstract. *As the technology and information develops, the greater the risk or the crime occurs. One of the most common crimes is counterfeiting and hijacking the information sent. Therefore, to maintain the confidentiality of such information requires a technology that can conceal information transmitted, such as cryptography. Based on the key used, the cryptographic algorithm consists of symmetric cryptography and asymmetric cryptography. One example of a symmetric key algorithm is the Advanced Encrypted Standard algorithm and the asymmetric key is Rivest-Shamir-Adleman. In this final project, built a digital image encryption and decryption system to analyze the performance of the two algorithms. Image type used is RGB image with different size. The results obtained in this system is, AES algorithm produce better cipher image but this algorithm require longer processing time compared with RSA algorithm. The performance of both methods is calculated by finding Mean Square Error (MSE) and Peak Signal to Noise Ratio (PSNR) values. From the average value of MSE and PSNR obtained, we found the perfect MSE and PSNR values on the RSA algorithm with MSE = 0 and PSNR = inf.*

Keyword: AES, RSA, MSE, PSNR

Abstrak. *Semakin berkembangnya teknologi dan informasi, semakin besar pula resiko atau kejahatan yang terjadi. Salah satu kejahatan yang sering terjadi adalah pemalsuan dan pembajakan informasi yang dikirim. Oleh sebab itu, untuk menjaga kerahasiaan informasi tersebut dibutuhkan sebuah teknologi yang dapat merahasiakan informasi yang dikirim, seperti kriptografi. Berdasarkan kunci yang digunakan, algoritma kriptografi terdiri dari kriptografi simetris dan kriptografi asimetris. Salah satu contoh algoritma kunci simetris adalah algoritma Advanced Encrypted Standard dan kunci asimetris adalah Rivest-Shamir-Adleman. Pada proyek akhir ini, dibangun sebuah sistem enkripsi dan dekripsi citra digital untuk menganalisis performa kedua algortima tersebut. Jenis citra yang digunakan yaitu citra RGB dengan berbagai ukuran yang berbeda. Hasil yang didapatkan pada sistem ini adalah, algoritma AES menghasilkan cipher image yang lebih baik namun algoritma ini membutuhkan waktu proses yang lebih lama dibandingkan dengan algoritma RSA. Kinerja kedua metode dihitung dengan mencari nilai Mean Square Error (MSE) dan Peak Signal to Noise Ratio (PSNR) citra. Dari rata-rata nilai MSE dan PSNR yang didapatkan, ditemukan nilai MSE dan PSNR yang sempurna pada algoritma RSA dengan nilai MSE = 0 dan PSNR = inf.*

Kata Kunci : AES, RSA, MSE, PSNR.

1. Pendahuluan

Dengan berkembang pesatnya teknologi, menjadikan internet sebagai wadah utama dalam perpesanan. Namun resiko dari kegiatan tersebut adalah terjadinya pembajakan dan kebocoran informasi. Maka dari itu, dibutuhkan sebuah teknologi untuk merahasiakan informasi tersebut, salah satunya yaitu kriptografi.

Kriptografi adalah ilmu mengenai teknik pengamanan atau penyembunyian komunikasi diatara dua pihak. Berdasarkan jenis kunci yang digunakan, algoritma kriptografi terbagi

menjadi 2 yaitu kriptografi kunci simetri dan kriptogrfi kunci asimetri. Dimana, kunci simetri ini menggunakan kunci yang sama untuk proses enkripsi dan dekripsi, sedangkan kunci asimetri ini menggunakan kunci yang berbeda untuk proses enkripsi dan dekripsi.

Pada tahun 1998, Electronic Frontier Fondation (EFF) telah berhasil memecahkan salah satu kriptografi kunci simetrik yang terkenal yaitu Data Encryption Standard dalam waktu 4-5 hari. Pada akhirnya DES dianggab sudah tidak aman lagi sehingga digantikan oleh Advanced Encryption Standard pada tahun 2001 [1]. Pada tahun sebelumnya, Massachusetts Institue of Technology mempatenkan sebuah algoritma kunci asimetrik bernam algoritma Rivest Shamir Adleman (RSA).

Dari kedua algoritma tersebut, pada proyek akhir ini dibagun sebuah sistem enkripsi menggunakan bahasa pemograman MATLAB. Jenis citra yang digunakan yaitu citra RGB dengan berbagai format file. Dari hasil enkripsi dan dekripsi kedua algoritma. nantinya akan dianalisa untuk mendapatkan sebuah algoritma yang lebih baik dalam proses enkripsi dan dekripsi citra digital.

2. Landasan Teori

2.1 Kriptografi

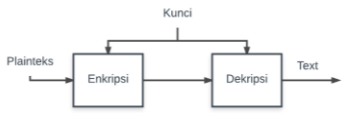
Kriptografi adalah ilmu dan seni untuk menjaga kerahasiaan pesan dengan cara menyandikannya ke dalam bentuk yang tidak dapat dimengerti lagi maknanya.

2.2 Algoritm Kriptografi

Algoritma kriptografi disebut juga *cipher* yaitu aturan untuk *enchipering* dan *dechipering* [2], atau fungsi matematika yang digunakan untuk enkripsi dan dekripsi. Beberapa *cipher* memerlukan algoritma yang berbeda untuk *enciphering* dan *dechiphering*.

2.3 Algoritma Simetri

Algoritma ini juga sering disebut dengan algoritma klasik, karena memakai kunci yang sama untuk kegiatan enkripsi dan dekripsinya [3]

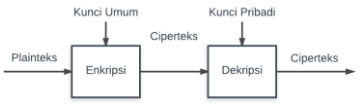


Gambar 1. Algoritma Simetri

2.4 Algoritma Non Simetri

Algoritma tak simetri sering juga disebut dengan algoritma kunci publik, dengan arti kata kunci yang digunakan untuk melakukan enkripsi dan dekripsinya berbeda. Pada algoritma tak simetri kunci terbagi menjadi 2 (dua) bagian:

1. Kunci umum (*public key*) adalah kunci yang dapat dan boleh diketahui oleh semua orang.
2. Kunci pribadi (private key) adalah kunci yang hanya dapat diketahui penerima dan bersifat rahasia.



Gambar 2. Algoritma Non Simetri

2.5 Algoritma Advanced Encryption Standard (AES)

Algorima Rijndael menggunakan substitusi, permutasi dan sejumlah putaran yang dikenakan pada tiap blok yang akan dienkripsi dekripsi [2]. Untuk setiap putarannya, Rijndael menggunakan kunci yang berbeda. Kunci setiap putaran disebut *round key*. Ukuran blok untuk algoritma Rijndael adalah 128 *bit* (16 *byte*).

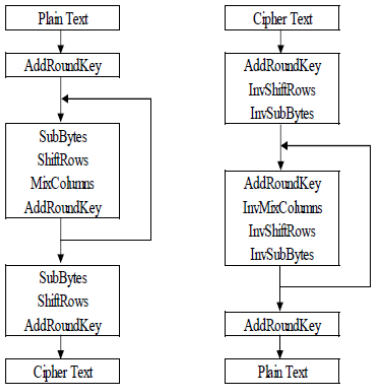
Rijndael mendukung panjang kunci 128 bit sampai 256 bit dengan step 32 bit [4]. Karena AES menetapkan panjang kunci adalah 128, 192, dan 256, maka dikenal AES-128, AES-192, dan AES-256.

	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 3. Panjang Kunci Rijndael

Algoritma Rijndael mempunyai 3 (tiga) parameter [2]:

1. *Plaintext* adalah array yang berukuran 16 *byte*, yang berisi data masukan.
2. *Ciphertext* adalah array yang berukuran 16 *byte*, yang berisi hasil enkripsi.
3. Kunci adalah array yang berukuran 16 *byte*, yang berisi kunci *cipher* (disebut juga *chiper key*).



Gambar 4. Diagram Proses Enkripsi dan Dekripsi Algortima AES

2.6 Algoritma Rivest Shamir Adleman

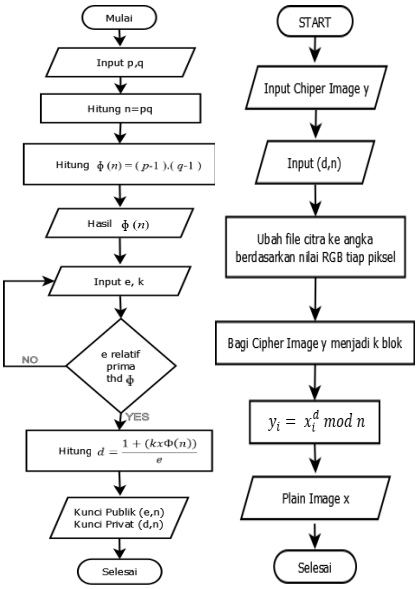
Algoritma RSA merupakan algoritma kunci publik yang mendasarkan keamanannya pada tingkat kesulitan dalam memfaktorkan bilangan yang besar menjadi factor-faktor prima [5]. Pemfaktoran dilakukan untuk memperoleh kunci privat. Selama pemfaktoran bilangan besar menjadi faktor-faktor prima yang ditemukan algoritma yang mangkus, maka selama itu pula keamanan algoritma RSA tetap terjamin [5].

Proses Pembuatan Kunci

Dalam membuat suatu sandi, RSA mempunyai cara kerja dalam membuat kunci publik dan kunci privat adalah sebagai berikut:

1. Pilih dua bilangan prima p dan q secara acak dengan ketentuan, $p \neq q$.
2. Hitung $n = pq$. Bilangan N disebut *parameter sekuriti*.
3. Hitung $\varphi = (p-1)(q-1)$.
4. Pilih bilangan bulat (*integer*) antara satu dan φ ($1 < e < \varphi$) yang tidak mempunyai faktor pembagi dari φ.
5. Hitung d hingga $d\,e \equiv 1 \pmod{\varphi}$ menggunakan cara *Extended Euclideon Algorithm*.

Setelah melalui cara ini, maka kita akan mendapatkan kunci privat. Kunci public terdiri dari n (modulus yang digunakan) dan e (eksponen public atau eksponen enkripsi). Sedangkan kunci privat terdiri dari n (modulus yang digunakan) dan d (eksponen pribadi atau eksponen dekripsi).

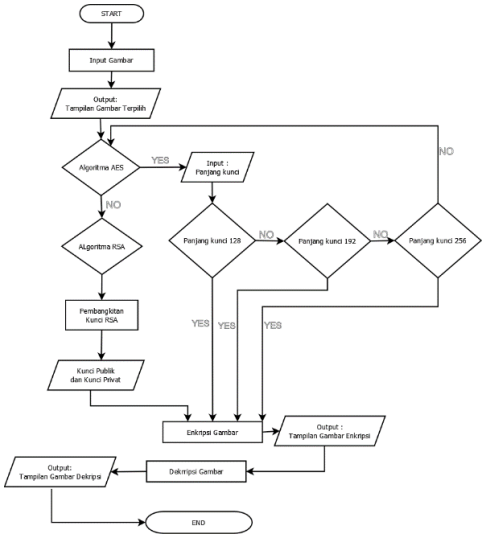


Gambar 5. Flowchart Proses Enkripsi dan Dekripsi ALgortima RSA

3. Rancangan Algoritma Dan Program

3.1 Algoritma Aplikasi

Asistem algoritma AES dan RSA untuk enkripsi dan dekripsi gambar dibanggun menggunakan Bahasa pemograman MATLAB. Pada enkripsi algoritma RSA, proses utama yang dilakukan yaitu *key generation* (pembangkitan kunci). Proses *key generation* dilakukan untuk mendapatkan pasangan kunci, yaitu *public key* dan *private key*. Pasangan kunci tersebut digunakan dalam proses enkripsi dan dekripsi. Kunci publik digunakan dalam proses enkripsi, dan kunci privat digunakan untuk proses dekripsi. Sedangkan pada algoritma AES, enkripsi citra membutuhkan panjang kunci. Panjang kunci ini dikelompokkan menjadi 3, AES-128, AES-192, dan AES-256. Angka-angka dibelakang AES menggambarkan panjang kunci yang digunakan pada tiap-tiap AES.



Gambar 6. Flowchart Aplikasi

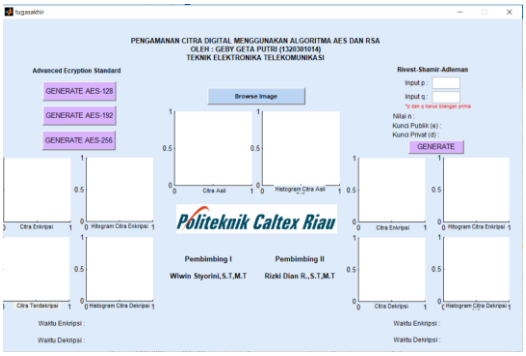
4. IMPLEMENTASI DAN PENGUJIAN PROGRAM

4.1 Pengujian Aplikasi

Pengujian aplikasi menjelaskan tentang cara pengoperasian program serta tahap-tahap yang perlu dilakukan user untuk menjalankan aplikasi enkripsi dan dekripsi file citra.

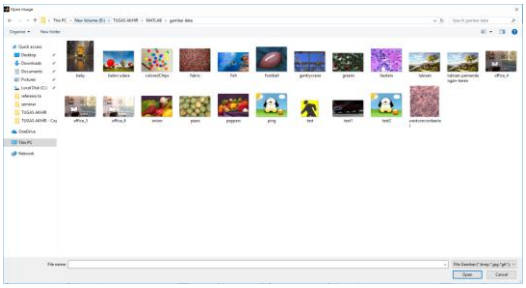
Tampilan Menu Utama

Tampilan menu utama aplikasi merupakan tampilan dimana untuk mulai menggunakan aplikasi yang terlihat pada gambar 7.



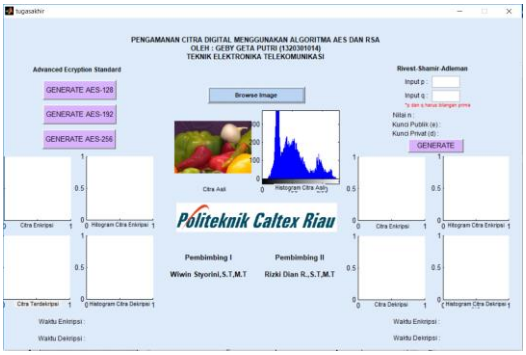
Gambar 7. Tampilan Aplikasi

User memilih *browse image* maka akan muncul open dialog file gambar yang digunakan untuk memilih file gambar pada *drive* penyimpanan dan kemudian menampilkan gambar asli yang terlihat pada gambar 8.



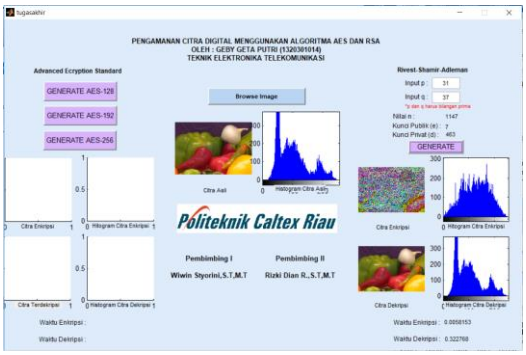
Gambar 8. Tampilan Open Image

Setelah gambar asli dipilih maka aplikasi akan menampilkan *plain image* beserta histogramnya seperti yang terlihat pada gambar 9.



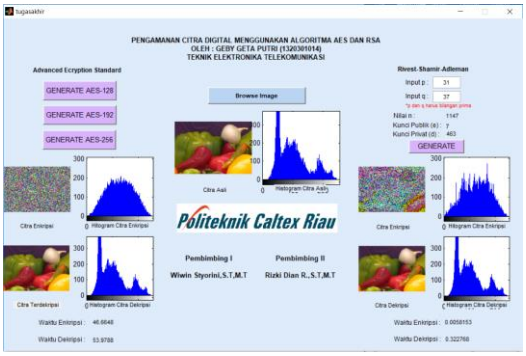
Gambar 9. Tampilan Plain Image

Untuk melakukan enkripsi dan dekripsi menggunakan algoritma RSA, *user* harus memasukan nilai prima p dan q pada *edit box* yang telah disediakan. Setelah user menekan *push button* '*generate*' maka sistem akan menghitung nilai n beserta kunci public dank unci privat dan menampilkan hasil enkripsi dan dekripsi beserta waktu proses dan histogranya.



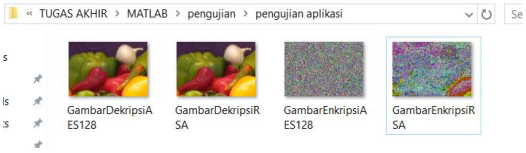
Gambar 10. Tampilan Hasil Proses Algoritma RSA

Untuk melakukan enkripsi dan dekripsi menggunakan algoritma AES dengan menggunakan *plain image* yang sama, *user* dapat memilih jenis AES yang akan digunakan dengan menekan salah satu tombol *push button* yang ada. Contohnya, user memilih jenis AES-128, maka sistem akan menampilkan hasil enkripsi dan dekripsi beserta waktu dan histogramnya.



Gambar 11. Tampilan Hasil Proses Algoritma AES

Gambar-gambar hasil enkripsi dan dekripsi pada aplikasi akan disimpan pada *folder* penyimpanan yang sama.



Gambar 12. Penyimpanan Gambar Hasil Enkripsi dan Dekripsi

4.2 Pangujian dan Analisis Hasil Pengujian

Pada uji coba ini, akan dilakukan dengan 2 skenario untuk setiap algoritma. Pada algoritma RSA, skenario pertama adalah dilakukan pengujian dengan berbagai pasangan kunci public dan kunci privat terhadap *plain image* yang sama, dan scenario kedua adalah dilakukan pengujian dengan berbagai ukuran dimensi gambar terhadap pasangan kunci yang sama.

Pengujian Algoritma RSA

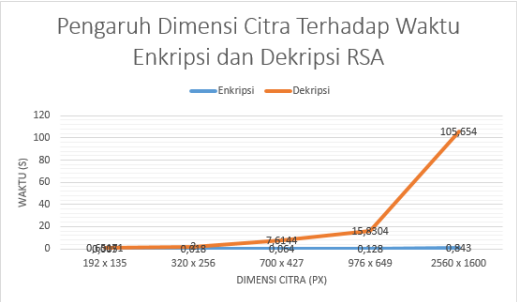
Tabel 1. Hasil Enkripsi dan Dekripsi RSA

No.	Citra Digital Asli	Citra Digital Terenkripsi	Citra Digital Terenkripsi	Waktu Dekripsi	Waktu Enkripsi
1.	Onion.png Dimensi 198 x 135 43.5 Kb	GambarEnkripsiRSA.png Dimensi 198 x 135 72 Kb	GambarDekripsiRSA.png Dimensi 198 x 135 44 Kb	0.651716	0.005506
2.	Football.jpg Dimensi 320 x 256 26.4 Kb	GambarEnkripsiRSA.png Dimensi 320 x 256 237 Kb	GambarDekripsiRSA.png Dimensi 320 x 256 150 Kb	2.04508	0.018014
3.	Lukisan.jpg Dimensi 700 x 427 69.9 Kb	GambarEnkripsiRSA.png Dimensi 700 x 427 818 Kb	GambarDekripsiRSA.png Dimensi 700 x 472 451 Kb	7.6144	0.064192
4.	Parasut.jpg Dimensi 976 x 649 105 Kb	GambarEnkripsiRSA.png Dimensi 976 x 649 1.388 Kb	GambarDekripsiRSA.png Dimensi 976 x 649 855 Kb	15.8304	0.128574

Tabel 2. Pengaruh Kunci Publik (e) dan kunci privat (d) terhadap waktu enkripsi dan dekripsi

No.	P	q	n	e	d	Waktu Enkripsi	Waktu Dekripsi
1.	3	7	21	5	5	0.00448	0.0041997
2.	79	31	2449	7	1003	0.005694	0.654452
3.	863	593	511759	3	340203	0.002341	218.56
4.	8191	2659	21779e+07	11	11874e+07	0.007511	7507.25

Dari tabel 2 terlihat bahwa besar kecilnya nilai kunci publik (e) mempengaruhi proses enkripsi, ini sesuai dengan rumus enkripsi $C = P^e \pmod n$. Dimana semakin besar nilai e maka akan semakin besar nilai pemangkatan dari P (*plain text*). Begitu juga dengan pengaruh kunci privat terhadap proses dekripsi, berdasarkan rumus dekripsi $P = C^d \pmod n$, semakin besar nilai d maka akan semakin besar nilai pemangkatan C dan akan semakin lama proses dekripsi berlangsung.

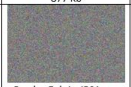


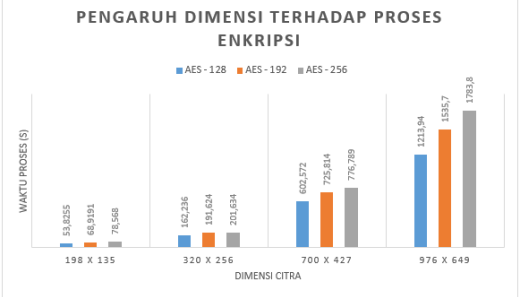
Gambar 13. Pengaruh Dimensi Citra Terhadap Waktu Enkripsi dan Dekripsi RSA

Lama proses enkripsi dan dekripsi juga dipengaruhi oleh ukuran dimensi citra yang digunakan. Terlihat pada diagram 1, semakin besar dimensi citra, maka akan semakin besar waktu yang dibutuhkan, hal ini dikarenakan oleh jumlah *pixel* yang ada juga semakin besar sehingga menambah waktu proses enkripsi dan dekripsi.

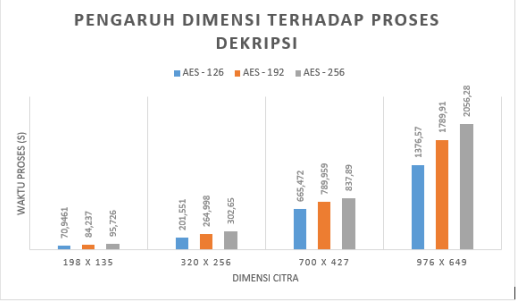
4.3 Pengujian algoritma AES

Tabel 3. Hasil Enkripsi dan Dekripsi AES

No.	Citra Digital Asli	Citra Digital Terenkripsi	Waktu Enkripsi	Citra Digital Terenkripsi	Waktu Dekripsi
1.	 Onion.png Dimensi 198 x 135 43.5 Kb	 GambarEnkripsiRSA.png Dimensi 198 x 135 78.6 Kb	53.8255s	 GambarEnkripsiRSA.png Dimensi 198 x 135 43.5 Kb	70.9461
2.	 Football.jpg Dimensi 320 x 256 26.4 Kb	 GambarEnkripsiRSA.png Dimensi 320 x 256 240 Kb	162.236	 GambarEnkripsiRSA.png Dimensi 320 x 256 43.5 Kb	201.511
3.	 Lukisan.jpg Dimensi 700 x 427 69.9 Kb	 GambarEnkripsiRSA.png Dimensi 700 x 427 877 Kb	602.572	 GambarEnkripsiRSA.png Dimensi 700 x 427 450 Kb	665,471
4.	 Parasut.jpg Dimensi 976 x 649 105 Kb	 GambarEnkripsiRSA.png Dimensi 976 x 649 1.81 MB	1213.94	 GambarEnkripsiRSA.png Dimensi 976 x 649 855 Kb	1376,91

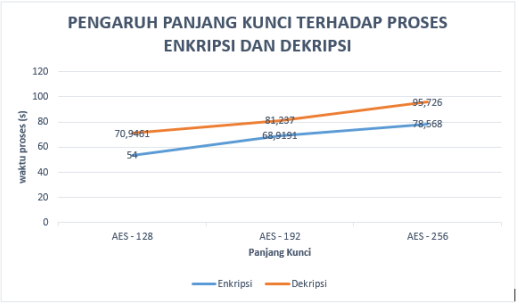


Gambar 14. Pengaruh Dimensi Gambar Terhadap Proses Enkripsi



Gambar 15. Pengaruh Dimensi Gambar terhadap Proses Dekripsi

Diagram 2 dan 3 menampilkan bagaimana pengaruh besar kecil dimensi suatu gambar terhadap proses enkripsi dan dekripsi. Sama halnya dengan algoritma RSA, besaran gambar sangat berpengaruh terhadap proses enkripsi dan dekripsi. Karena, semakin besarnya dimensi gambar, semakin banyak *pixel* yang akan diolah.



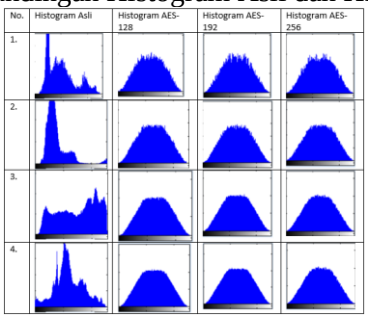
Gambar 16. Pengaruh Panjang Kunci Terhadap Proses Enkripsi dan Dekripsi

Berdasarkan diagram 4 terlihat bahwa pada proses enkripsi dan dekripsi dengan panjang kunci AES-256 membutuhkan waktu yang lebih lama dibandingan AES-128 dan AES-192. Hal ini disebabkan karena banyaknya round yang terjadi pada AES-256 lebih lama dibandingkan dengan lainnya.

4.4 Pengujian Kualitas Enkripsi

Untuk melihat kualitas enkripsi dilakukan pengujian terhadap histogram citra. Teknik ini digunakan untuk melihat kesesuaian distribusi warna antara plain image dengan cipher image. Dari hasil pengujian yang ditampilkan pada tabel 4, histogram kedua image memiliki distibusi keragaman dan perbedaan yang signifikan. Dengan adanya perbedaan yang signifikan ini dapat dikatakan bahwa *cipher image* tidak dapat memberikan informasi apa-apa mengenai plain image sehingga dapat melindungi algoritma dari serangan.

Tabel 4. Perbandingan Histogram Asli dan Histogram Citra



4.5 Pengujian Kualitas Dekripsi

Tabel 5. Nilai MSE dan PSNR Hasil Dekripsi Algoritma RSA

No.	Nama File	Mean Squared Error			Peak Signal to Noise Ratio (db)		
		R	G	B	R	G	B
1.	Onion.png	0	0	0	Inf	Inf	Inf
2.	Football.jpg	0	0	0	Inf	Inf	Inf
3.	Lukisan.jpg	0	0	0	Inf	Inf	Inf
4.	Parasut.jpg	0	0	0	Inf	Inf	Inf

Tabel 6. Nilai MSE dan PSNR Hasil Dekripsi Algoritma AES

No.	Nama File	Mean Squared Error			Peak Signal to Noise Ratio (db)		
		R	G	B	R	G	B
1.	Onion.png	0	0	9.63	Inf	Inf	31.894
2.	Football.jpg	0	0	0	Inf	Inf	Inf
3.	Lukisan.jpg	0	0	0	Inf	Inf	Inf
4.	Parasut.jpg	0	0	0.94	Inf	Inf	56.031

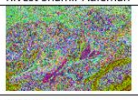
Kualitas citra hasil dekripsi dapat dilihat dari nilai PSNR (Peak Signal Noise Ratio) dan MSE (Mean Squared Error). MSE adalah nilai error kuadrat rata-rata antara citra asli dengan citra asli dan dekripsi. Sedangkan PSNR adalah perbandingan antara nilai maksimum dari kualitas citra asli dan citra yang sudah mengalami proses dekripsi. Secara teori, nilai PSNR dapat dihitung dengan rumus.

$$PSNR = 10log_{10}\left(\frac{{I_{max}}^2}{MSE}\right)$$

PSNR sering dinyatakan dalam skala logaritmik dalam decibel (dB). I_{max} merupakan nilai intensitas tertinggi. Ini menandakan, semakin besar nilai PSNR maka semakin bagus hasil dekripsi, dan semakin mirip citra dekripsi dengan citra aslinya. Ketika MSE yang terukur bernilai 0, maka dapat dikatakan bahwa kedua citra yang dibandingkan mirip satu sama lain. Jika MSE bernilai 0 maka otomatis PSNR bernilai tak terhingga (infinity). Dan jika MSE bernilai 0 dan PSNR tak terhingga, maka dapat dikatakan tidak ada data yang lose pada proses dekripsi atau dapat dikatakan bahwa gambar dekripsi sama dengan gambar asli.

4.6 Perbandingan Kedua Algoritma

Tabel 7. Perbandingan Kinerja Algoritma AES dan RSA

Parameter Perbandingan	Rivest Shamir Adleman	Advanced Encryption Standard
Hasil enkripsi		
Rata-rata waktu enkripsi	0,094s	508,14s
Rata-rata waktu dekripsi	6,535s	578,63s
MSE	0	9,70
PSNR (dB)	Inf	36,56

Pada tabel 7, dapat diketahui perbandingan kinerja pada kedua algoritma. Algoritma AES menghasilkan kualitas hasil enkripsi yang lebih baik dibandingan dengan algoritma RSA namun membutuhkan waktu yang lebih lama. Algoritma RSA memiliki nilai MSE rata-rata senilai 0 dan PSNR senilai inf, ini menandakan bahwa tidak terjadi perubahan atau error pada proses dekripsi sedangkan algoritma AES memiliki nilai MSE rata-rata yang lebih besar daripada RSA yaitu 9,7 dengan PSNR 36,56dB.

5. Kesimpulan dan Saran

Kesimpulan

Setelah melakukan pengujian dan analisis, maka dapat diambil beberapa kesimpulan sebagai berikut :

1. Proses enkripsi dan dekripsi pada algoritma RSA dipengaruhi oleh pasangan kunci. Dimana, semakin besar kunci publik maka akan semakin lama proses enkripsi, dan semakin besar kunci privat maka akan semakin lama proses dekripsi.
2. Proses enkripsi dan dekripsi pada algoritma AES dipengaruhi oleh panjang kunci. Dimana semakin panjang kunci yang digunakan maka akan semakin banyak putaran yang dilalui dan semakin lama proses enkripsi dan dekripsi berlangsung.
3. Besarnya dimensi *plain image* yang digunakan mempengaruhi waktu enkripsi dan dekripsi. Semakin besar dimensi citra, maka akan semakin besar waktu yang dibutuhkan, hal ini dikarenakan oleh jumlah pixel yang ada juga akan semakin besar sehingga menambahkan waktu proses enkripsi dan dekripsi.
4. Algortima RSA menghasilkan nilai MSE=0 dan PSNR=inf. Ini menandakan bahwa tidak terjadi perubahan dan error antara gambar asli dengan gambar hasil dekripsi. Sedangkan algoritma AES menghasilkan nilai MSE rata-rata = 9,7 dengan PSNR = 36,56 dB.
5. Algoritma RSA lebih unggul pada kualitas dekripsi dan kecepatan proses enkripsi dan dekripsi, sedangkan algoritma AES lebih unggul pada kualitas enkripsi.

5.1 Saran

Dari penelitian yang telah dilakukan, penulis memberikan beberapa saran untuk penelitian selanjutnya.

1. Untuk dapat melakukan enkripsi dan dekripsi pada file suara atau video.
2. Melakukan enkripsi dan dekripsi dengan menggabungkan kedua algoritma.

Daftar pustaka

[1] Nicholas G. McDonald, “Past, Present, and Future Methods of Cryptography and Data Encryption”
[2] Ariyus, Dony, “Penganntar Ilmu KRIPTOGRAFI”: Yogyakarta
[3] Dafid, “Kriptografi Kunci Simetris dengan Menggunakan Algoritma Crypton”:Palembang, 2006
[4] Rifqi Azhar Nugraha, “Advanced Encryption Standard”, 2009
[5] Hersatoto Listiayono, “Implementasi Algortima Kunci Public Pada Algoritma RSA”, 2009