Enkripsi Gambar Berbasis Grid menggunakan Kriptografi Berbasis Kode

Dian Anggoro Putro Bhagaskoro[1], Ari Moesriami Barmawi[2]

[1,2]Fakultas Informatika, Universitas Telkom, Bandung
[1]bhagaskoro@students.telkomuniversity.ac.id, [2]mbarmawi@telkomuniversity.ac.id,

Abstrak

Beberapa tahun terakhir, keamanan gambar merupakan masalah besar dalam komunikasi data melalui jaringan yang tidak aman. Dalam proposal ini, penulis ingin membandingkan teknik untuk mengamankan gambar antara teknik McEliece Cryptography dan RSA. Gambar akan diubah menjadi blok-blok dan setiap blok akan dibagi menjadi beberapa grid dengan transformasi, kemudian akhirnya kedua teknik tersebut diterapkan pada grid untuk mengamankan gambar.

Kata Kunci: McEliece Cryptography, RSA, Enkripsi Gambar, Dekripsi Gambar

Abstract

In recent years, image security is a big problem in data communication over insecure networks. In this proposal, the author wants to compare techniques to secure images between McEliece Cryptography and RSA techniques. The image will be transformed into blocks and each block will be divided into grids with transformations, then finally the two techniques are applied to the grid to secure the image.

Keywords: McEliece Cryptography, RSA, Image Encryption, Image Decryption

1. Pendahuluan

Pada bab pendahuluan, penulis akan memaparkan latar belakang pembuatan tugas akhir. Setelah itu, penulis juga memberikan rumusan masalah dan tujuan proposal, serta memaparkan metodologi penyelesaian tugas akhir.

1.1. Latar Belakang

Saat ini, gambar sering digunakan untuk tukar-menukar informasi pada jaringan internet. Pada jaringan internet yang tidak aman, diperlukan teknik untuk mengamankan gambar (*confidentiality*) agar tidak dicuri oleh pihak yang tidak ikut dalam komunikasi data.

Salah satu teknik yang digunakan untuk mengamankan gambar adalah RSA [2]. Teknik RSA diperkenalkan pada tahun 1978 oleh R. Rivest, A. Shamir, dan L. Adleman dengan memanfaatkan teori bilangan dalam melakukan pengamanan data. Karena RSA memerlukan waktu komputasi yang sangat tinggi untuk melakukan enkripsi dan dekripsi, maka akan sangat tidak efisien untuk enkripsi pada gambar.

1.2. Perumusan Masalah

Proses enkripsi pada gambar menggunakan RSA memerlukan waktu yang tinggi, sehingga diperlukan solusi untuk mengamankan gambar dengan *McEliece Cryptography*.

1.3. Tujuan

Tujuan pada penelitian ini adalah untuk mempercepat permasalahan proses enkripsi dan dekripsi pada gambar menggunakan *McEliece Cryptography.*

1.4. Metodologi Penyelesaian Masalah

Metode yang digunakan dalam menyelesaikan masalah-masalah di dalam penelitian Tugas Akhir ini terdiri atas 5 tahapan, antara lain sebagai berikut. Langkah pertama yatiu melakukan studi literatur, yaitu dengan melakukan pencarian dan pembelajaran referensi dan sumber-sumber yang berkaitan dengan RSA *cryptosystem* dan McEliece cryptosystem. Setelah itu, akan dilakukan perancangan model. Sistem yang akan dibangun merupakan sebuah sistem yang membandingkan waktu komputasi enkripsi gambar antara algoritma RSA dan algoritma McEliece. Terdapat dua proses utama dalam sistem ini, yaitu proses enkripsi gambar dan dekripsi gambar. Pada proses enkripsi gambar dilakukan proses enkripsi gambar dengan algoritma RSA dan algoritma McEliece. Sedangkan pada proses dekripsi gambar dilakukan proses dekripsi gambar dengan algoritma RSA dan

algoritma McEliece. Setelah sistem dibangun, dilakukan implementasi metode RSA dan McEliece dalam proses enkripsi dan dekripsi gambar. Gambar yang dipilih adalah gambar-gambar berukuran 256x256 dan berjumlah 60 gambar. Dari 60 gambar, akan dibagi menjadi 3 kumpulan gambar, yaitu 20 gambar dengan histogram tengah, 20 gambar dengan histogram kiri, dan 20 gambar dengan histogram kanan [3]. Sistem akan di implementasikan dengan bahasa pemrograman Matlab. Kemudian, penulis melakukan pengujian terhadap sistem yang telah dibuat. Pengujian dilakukan dengan beberapa parameter, antara lain pengujian hasil enkripsi dari gambar dan pengujian terhadap waktu komputasi algoritma. Kemudian akan dilakukan analisis terhadap data hasil pengujian. Kemudian, penulisan buku tugas akhir dibuat untuk penulisan dokumentasi dari proses pengumpulan data sampai proses analisis pengujuan hasil.

1.5. Organisasi Tulisan

Setelah bagian Pendahuluan, bab selanjutnya adalah Tinjauan Pustaka, dimana penulis memberikan uraian singkat tentang teori-teori yang berkaitan dengan tugas akhir, antara lain RSA *Cryptosystem* dan McEliece *Cryptosystem*. Kemudian, penulis menjelaskan sistem yang dibangun untuk pengujian tugas akhir. Setelah sistem dibangun, penulis melakukan evaluasi terhadap hasil pengujian dengan membandingkan waktu komputasi antara proses enkripsi dan dekripsi dengan RSA *Cryptosystem* dan proses enkripsi dan dekripsi dengan McEliece *Cryptosystem*.

Pada akhirnya, penulisan memberikan kesimpulan dari serangkaian proses pengujian dan analisis yang telah dilakukan serta saran guna mengembangkan dan menyempurnakan penelitian pada tugas akhir ini.

2. Studi Terkait

Pada bab ini, penulis memberikan uraian singkat tentang teori-teori yang berkaitan dengan tugas akhir ini, antara lain Public-Key Cryptography, RSA, dan McEliece Cryptography. Selain itu, penulis juga memberikan metode pada jurnal yang dijadikan acuan, yaitu Enkripsi Gambar Berbasis Grid menggunakan RSA, dan metode teknik kriptografi yang ada pada bab ini.

2.1. RSA

Pada tahun 1978, R. Rivest, A. Shamir, dan L. Adleman membangun Kriptografi kunci publik, termasuk pembuatan kunci dan kunci publik, yang keamanannya terletak pada kesulitan teori bilangan [4]. Kriptografi ini, dikenal dengan akronim dari nama penulis, RSA. Kriptografi RSA masih digunakan sampa saat ini, dimana system tersebut digunakan dalam aplikasi kriptografi pada perbankan, dan keamanan e-mail di e-commerce.

2.1.1. Pembangktian Kunci pada RSA

Sebelum memulai proses enkripsi dan dekripsi, pengirim menentukan kunci yang akan digunakan terlebih dahulu. Untuk pembangkitan kunci pada RSA, perlu dilakukan tahap-tahap sebagai berikut.

a. Pilih dua bilangan prima berbeda p dan q yang besar agar tidak mudah difaktorkan.
b. Hitung $n = pq$. Setelah n diperoleh, bilangan tersebut akan digunakan sebagai kunci publik
c. Hitung $\phi(n) = \phi(p).\phi(q) = (p-1)(q-1)$. ϕ adalah fungsi Euler.
d. Pilih bilangan bulat d sedemikian sehingga $1 < d < \phi(n)$ dan $gcd\,(d, \phi(n)) = 1$. d dan $\phi\,(n)$ adalah bilangan relatif prima. Setelah d diperoleh, angka tersebut akan digunakan sebagai kunci privat dari kedua pihak dan digunakan untuk dekripsi pesan.
e. Tentukan e dengan menghitung $e \equiv d^{-1}\ (mod\ \phi(n))$. e adalah invers perkalian modular dari d pada modulus $\phi(n)$, sehingga $d.e \equiv 1\ (mod\ \phi(n))$. Setelah itu, e digunakan sebagai kunci publik, sehingga (n, e) adalah kunci untuk enkripsi pesan.

2.1.2. Enkripsi dan Dekripsi dengan RSA

Enkripsi dilakukan dengan cara membagi pesan *M* dengan beberapa urutan blok $X_1, X_2, \ldots, X_n$ dimana setiap X_i memenuhi $0 \leq X_i < n$. Kemudian, proses enkripsi dilakukan dengan cara

$$\boldsymbol{C \equiv E(M)}$$
$$\boldsymbol{C \equiv M^e\ (mod\ n)}$$

dimana $\boldsymbol{C}$ adalah *Ciphertext* (Pesan yang telah dienkripsi), $\boldsymbol{E(M)}$ adalah fungsi enkripsi, $\boldsymbol{e}$ adalah kunci publik untuk enkripsi pesan, dan $\boldsymbol{M}$ adalah pesan asli.

Setelah C diterima, penerima melakukan proses dekripsi menggunakan kunci privat d yang telah diperoleh dari pembangkitan kunci. Dekripsi dilakukan dengan cara

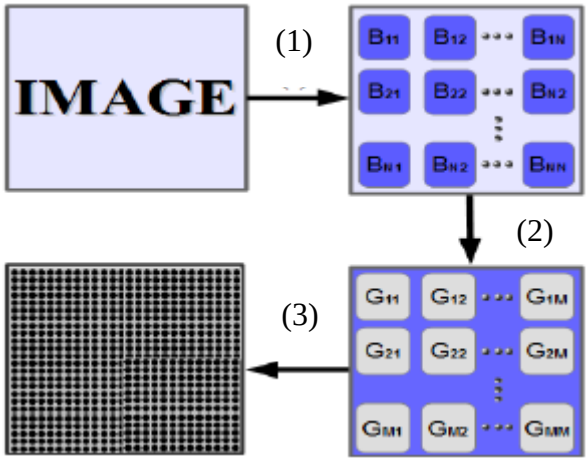
$$\boldsymbol{M \equiv D(C)}$$
$$\boldsymbol{M \equiv C^d\ (mod\ n)}$$

dimana $\boldsymbol{M}$ adalah pesan asli, $\boldsymbol{D(C)}$ adalah fungsi dekripsi, $\boldsymbol{d}$ adalah kunci privat untuk dekripsi *Ciphertext*, dan $\boldsymbol{C}$ adalah *Ciphertext*. [2]

2.2. Enkripsi Gambar Berbasis Grid Menggunakan RSA

Pada bagian ini, akan dipaparkan tahap-tahap enkripsi gambar dengan RSA, yaitu membagi gambar menjadi beberapa blok, mengacak blok-blok gambar, membagi setiap blok pada gambar menjadi beberapa grid, kemudian setiap grid pada semua blok diacak kembali, dan terakhir dilakukan enkripsi gambar berdasarkan hasil pengacakan piksel pada gambar.

Enkripsi gambar dengan RSA pada umumnya digambarkan dengan diagram sebagai berikut.



Gambar 2.1 Alur Enkripsi Gambar Berbasis Grid dengan RSA

Sebelum proses enkripsi dimulai, hal pertama yang harus dilakukan adalah membagi gambar menjadi beberapa blok. Kemudian, blok tersebut diacak dengan cara berdasarkan algortima enkripsi di bawah. Setiap blok memiliki jumlah baris dan kolom yang sama. Blok-blok pada gambar tersebut ditulis dengan persamaan matematika sebagai berikut.

$$frame = \sum_{l=0}^{l=N} \sum_{m=0}^{m=N} B_{lm}, \quad (1)$$

dimana l = baris blok, m = kolom blok, B = blok pada gambar, N = jumlah baris dan kolom.

Setelah gambar dibagi menjadi beberapa blok, setiap blok dibagi menjadi beberapa grid dan posisi semua grid di dalam setiap blok diacak dengan berdasarkan algortima enkripsi di bawah. Grid-grid pada suatu blok ditulis dengan persamaan matematika sebagai berikut.

$$B_{11} = \sum_{n=0}^{n=M} \sum_{r=0}^{r=M} G_{nr}, \quad (2)$$

dimana n = baris grid, r = kolom grid, G = grid pada blok, M = jumlah baris dan kolom.

Setelah blok dibagi menjadi beberapa grid, kemudian setiap grid pada semua blok dibagi menjadi beberapa pixel dan posisi semua pixel di dalam setiap grid diacak berdasarkan algortima enkripsi di bawah. Pixel-pixel pada suatu Grid ditulis dengan persamaan matematika sebagai berikut.

$$G_{11} = \sum_{x=0}^{x=K} \sum_{y=0}^{y=K} p_{xy}, \quad (3)$$

dimana x = baris pixel, y = kolom pixel, p = pixel pada grid, K = jumlah baris dan kolom.

Setelah pixel-pixel teracak, setiap pixel dienkripsi dengan sistem RSA sesuai dengan algoritma enkripsi di bawah dan menghasilkan persamaan berikut.

Algoritma yang dilakukan untuk pengacakan gambar dan enkripsi gambar ditulis sebagai berikut.

Algoritma 1 Enkripsi Gambar dengan RSA

Input: Original Image

1: Membagi gambar asli menjadi beberapa blok dengan ukuran piksel yang sama
2: for each block do
3: tukar blok dengan skema $B_{00} \rightarrow B_{MM}; B_{01} \rightarrow B_{M-1M}$*; dan seterusnya;*
untuk setiap blok B_{nr}*, dimana* $n = 0..M$ *dan* $r = 0..M;$
4: Membagi setiap blok menjadi beberapa grid dengan ukuran piksel yang sama
5: for each grid do
6: tukar grid dengan skema $G_{00} \rightarrow G_{KK}; G_{01} \rightarrow G_{K-1K}$*; dan seterusnya;*
untuk setiap grid G_{xy}*, dimana* $x = 0..K$ *dan* $y = 0..K;$
7: for each pixel do
8: tukar piksel dalam setiap grid dengan skema
$P_{00} \rightarrow P_{LL}; P_{01} \rightarrow P_{L-1L}$*; dan seterusnya;*
untuk setiap P_{lm}*, dimana* $l = 0..L$ *dan* $m = 0..L;$
end for
end for
end for
9: Enkripsi setiap piksel dengan RSA cryptosystem: $P_{ij} \rightarrow P'_{ij};$
10: Satukan kembali gambar setelah enkripsi

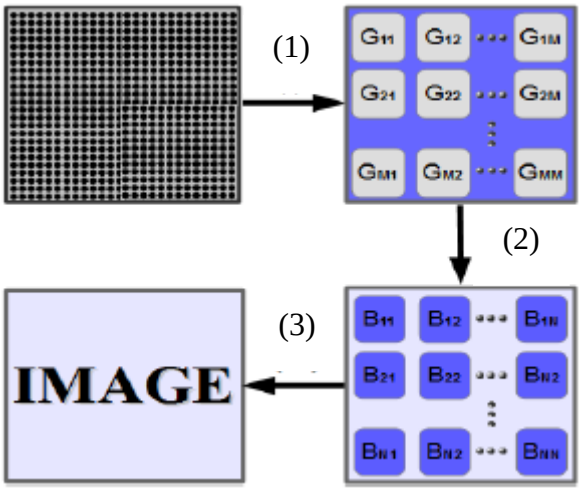
Output: Encrypted Image

Ket:

$p_{ij} = pixel\ sebelum\ enkripsi, p'_{ij} = pixel\ setelah\ enkripsi.$

2.2.1. Dekripsi Gambar

Dekripsi gambar dengan RSA adalah kebalikan dari proses enkripsi dan digambarkan dengan diagram sebagai berikut.



Gambar 2.2 Alur Dekripsi Gambar Berbasis Grid dengan RSA

Pada proses dekripsi, setiap pixel didekripsi menjadi pixel asli. Setelah setiap pixel didekripsi, setiap pixel pada setiap grid dikembalikan pada urutan semula dan ditulis dengan persamaan sebagai berikut.

$$G_{11} = \sum_{x=0}^{x=K} \sum_{y=0}^{y=K} p_{xy}, \quad (1)$$

dimana x = baris pixel, y = kolom pixel, p = pixel pada grid yang telah didekripsi, K = jumlah baris dan kolom.

Setelah setiap grid didekripsi, setiap grid pada setiap blok dikembalikan pada urutan semula dan ditulis dengan persamaan sebagai berikut.

$$B_{11} = \sum_{n=0}^{n=M} \sum_{r=0}^{r=M} G_{nr}, \quad (2)$$

dimana n = baris grid, r = kolom grid, G = grid yang telah didekripsi pada blok, M = jumlah baris dan kolom.

Setelah setiap blok didekripsi, setiap blok pada gambar dikembalikan pada urutan semula hingga kembali menjadi gambar asli dan ditulis dengan persamaan sebagai berikut.

$$frame = \sum_{l=0}^{l=N} \sum_{m=0}^{m=N} B_{lm}, \quad (3)$$

dimana l = baris blok, m = kolom blok, B = blok yang didekripsi pada gambar, N = jumlah baris dan kolom.

Algoritma yang dilakukan untuk penyatuan gambar menjadi gambar asli ditulis sebagai berikut.

Algoritma 2 Dekripsi Gambar dengan RSA

Input: Encrypted Image

1: Dekripsi setiap piksel dengan RSA cryptosystem: $P'_{ij} \rightarrow P_{ij}$*;*
2: Membagi gambar enkripsi menjadi beberapa blok yang sesuai dengan ukuran blok pada proses enkripsi.
3: Membagi setiap blok menjadi beberapa grid menjadi beberapa grid yang sesuai dengan ukuran grid pada proses enkripsi.
4: for each pixel in a grid do
5: tukar pixel dengan skema $P_{LL} \rightarrow P_{00}; P_{L-1L} \rightarrow P_{01}$*; dan seterusnya; untuk setiap piksel* P_{lm}*, dimana* $l = 0..L$ *dan* $m = 0..L$*;*
6: for each grid in a block do
7: tukar grid dengan skema $G_{KK} \rightarrow G_{00}; G_{K-1K} \rightarrow G_{01}$*; dan seterusnya; untuk setiap grid* G_{xy}*, dimana* $x = 0..K$ *dan* $y = 0..K$*;*
8: for each block do
9: tukar blok dengan skema $B_{MM} \rightarrow B_{00}; B_{M-1M} \rightarrow B_{01}$*; dan seterusnya; untuk setiap blok* B_{nr}*, dimana* $n = 0..M$ *dan* $r = 0..M$*;*
end for
end for
end for
10: Satukan kembali gambar setelah dekripsi dan pertukaran pixel

Output: Decrypted Image

Ket:

$p'_{ij} = pixel\ sebelum\ dekripsi, p_{ij} = pixel\ setelah\ dekripsi.$

2.3. McEliece Cryptography

Pada umumnya, keamanan sistem kriptografi kunci publik didasarkan pada kerumitan pada faktorisasi bilangan bulat atau sulitnya menemukan bilangan logaritma diskret. Robert McEliece adalah orang pertama yang mempunyai ide tentang sistem kriptografi kunci publik yang keamanannya didasarkan pada koreksi *error* pada kode [3,5].

Sistem kriptografi McEliece didasarkan pada pemilihan kode khusus yang memiliki algoritma decoding yang efisien sehingga proses dekripsi menjadi lebih cepat. McEliece menetapkan untuk menggunakan salah satu kelas kode yaitu kode hamming. Selain mempunyai algoritma decoding yang efisien, kode hamming juga dapat mengoreksi semua *error* berbobot satu yang tertambahkan pada kode sehingga proses penentuan kunci lebih cepat. Proses dekripsi pada sistem kriptografi McEliece didasarkan pada teori deteksi dan koreksi *error* pada kode.

Sistem kriptografi McEilece menjaga kerahasiaan. pesan dengan tiga cara, yaitu dengan menyembunyikan kunci pribadi, mengalikan matriks generator untuk kode hamming dengan matriks sembarang yang memiliki invers dan matriks permutasi, serta menambahkan vektor *error* pada pesan yang akan dikirimkan. Proses-proses dalarn sistem kriptografi McEliece yang melibatkan matriks-matriks berukuran besar dapat diselesaikan dengan cepat dengan menggunakan program simulasi sistem kriptografi McEliece

2.3.1. Pembangkitan Kunci

Untuk membangkitkan kunci pada sistem kriptografi McEliece, perlu dilakukan tahap-tahap sebagai berikut.

1. Alice memilih kode linear biner $C(n,k)$ yang mampu memperbaiki *error* t. Kode ini harus mampu berperan sebagai algoritma *decoding* yang efisien, seperti *goppa code* dan membangkitkan matriks berukuran $k\ x\ n$ matriks pembangkit G untuk kode C.
2. Alice memilih secara acak matriks biner S yang memiliki invers berukuran $k\ x\ k$.
3. Alice memilih secara acak matriks permutasi P berukuran $n\ x\ n$.
4. Alice menghitung matriks $\hat{G}$ dengan ukuran $k\ x\ n$, yaitu $\hat{G} = SGP$.
5. Kunci publik Alice adalah $(\hat{G}, t)$ dan kunci privatnya adalah (S, G, P).

2.3.2. Proses Enkripsi

Misalkan Bob ingin mengirim pesan ke Alice dengan kunci publik $(\hat{G}, t)$. Bob melakukan enkripsi dengan tahap-tahap sebagai berikut.

1. Bob melakukan encoding terhadap pesan m menjadi deretan biner dengan panjang k.
2. Bob menghitung vektor $c' = m\hat{G}$.
3. Bob membangkitkan vektor e sebesar n-bit secara acak yang mengandung t angka 1, agar hasil enkripsi semakin acak dan berdasarkan kode goppa yang bisa memperbaiki *error* hingga *t* [8]. ($t = weight$)
4. Bob menghitung pesan rahasia $c = c' + e$.

Contoh Enkripsi dengan McEliece Cryptosystem

Misalkan pengirim ingin mengirimkan pesan m berupa satu piksel dari suatu gambar bernilai 14. Proses pertama yang dilakukan sebelum melakukan enkripsi adalah pembangkitan kunci. Pembangkitan kunci dilakukan dengan tahap-tahap sebagai berikut.

1. Misalkan nilai $k = 4$ dan $n = 7$.
2. Pilih sembarang matriks A berdimensi $k\ x\ (n-k)$ atau $4\ x\ 3$.

$$A = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

3. Pilih matriks generator berdimensi $k\ x\ n$ dan $G = [I_k \mid A]$.

$$G = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}$$

4. Pilih matriks non-singular S berdimensi $k\ x\ n$ atau $4\ x\ 4$.

$$S = \begin{bmatrix} 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 \end{bmatrix}$$

5. Pilih matriks permutasi P berdimensi $n\ x\ n$ atau $7\ x\ 7$.

$$P = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}$$

6. Simpan matriks S, G, dan P sebagai *private key*.
7. Hitung $G' = SGP$, yang merupakan *public key*.

$$G' = S.G.P = \begin{bmatrix} 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 & 0 & 1 \end{bmatrix}$$

Selanjutnya, proses enkripsi pesan dilakukan dengan tahap-tahap sebagai berikut.

1. Pesan piksel '14' diubah menjadi bilangan biner, sehingga menjadi 1110.
2. Pilih sembarang vektor $e = [1 \quad 0 \quad 1 \quad 1 \quad 1 \quad 0 \quad 1]$ berdimensi $1\ x\ n$ atau $1\ x\ 7$.
3. Lakukan proses enkripsi untuk memperoleh *ciphertext* c dengan cara:

$$c = m.G' \oplus e, \text{ sehingga}$$

$$c = [1 \quad 1 \quad 1 \quad 0].\begin{bmatrix} 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 & 0 & 1 \end{bmatrix} \oplus [1 \quad 0 \quad 1 \quad 1 \quad 1 \quad 0 \quad 1]$$

$$= [1 \quad 0 \quad 0 \quad 1 \quad 1 \quad 1 \quad 0]$$

Kemudian, pesan berupa *ciphertext* c dikirim ke penerima pesan.

2.3.3. Proses Dekripsi

Misalkan Alice telah menerima pesan rahasia c dari Bob. Alice perlu melakukan dekripsi dengan tahap-tahap sebagai berikut.

1. Alice menghitung (P^{-1}), invers dari P.
2. Alice menghitung $\hat{c} = c(P^{-1})$.
3. Alice menggunakan algoritma decoding untuk kode C untuk proses decoding $\hat{c}$ menjadi $\hat{m}$.
4. Alice menghitung $m = \hat{m}S^{-1}$.

Contoh Dekripsi dengan McEliece Cryptosystem

Berdasarkan contoh enkripsi yang telah dipaparkan, misalkan penerima telah menerima pesan *ciphertext* $c = [1 \quad 0 \quad 0 \quad 1 \quad 1 \quad 1 \quad 0]$ dari pengirim. Penerima perlu melakukan dekripsi dengan tahap-tahap sebagai berikut.

1. Menghitung (P^{-1}), invers dari P.

$$P^{-1} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}$$

2. Menghitung $\hat{c} = c(P^{-1})$, sehingga

$$\hat{c} = [1 \quad 0 \quad 0 \quad 1 \quad 1 \quad 1 \quad 0].\begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} = [1 \quad 1 \quad 1 \quad 1 \quad 0 \quad 0 \quad 0]$$

3. Penerima menggunakan algoritma decoding untuk kode c untuk proses decoding $\hat{c}$ menjadi $\hat{m}$ menggunakan kode Hamming, dan diperoleh $\hat{m} = [0 \quad 1 \quad 0 \quad 0 \quad 1 \quad 0 \quad 1]$.
4. Kemudian dari $\hat{m}$, diambil 4 digit pertamanya, sehingga menjadi $\hat{m} = [0 \quad 1 \quad 0 \quad 0]$.
5. Setelah itu, untuk mendapatkan pesan asli, pengirim menghitung $m = \hat{m}S^{-1}$. Sehingga diperoleh

$$S^{-1} = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$m = \hat{m}S^{-1} = [0 \quad 1 \quad 0 \quad 0].\begin{bmatrix}1 & 0 & 0 & 1\\1 & 1 & 1 & 0\\1 & 0 & 1 & 0\\0 & 0 & 1 & 0\end{bmatrix} = [1 \quad 1 \quad 1 \quad 0]$$

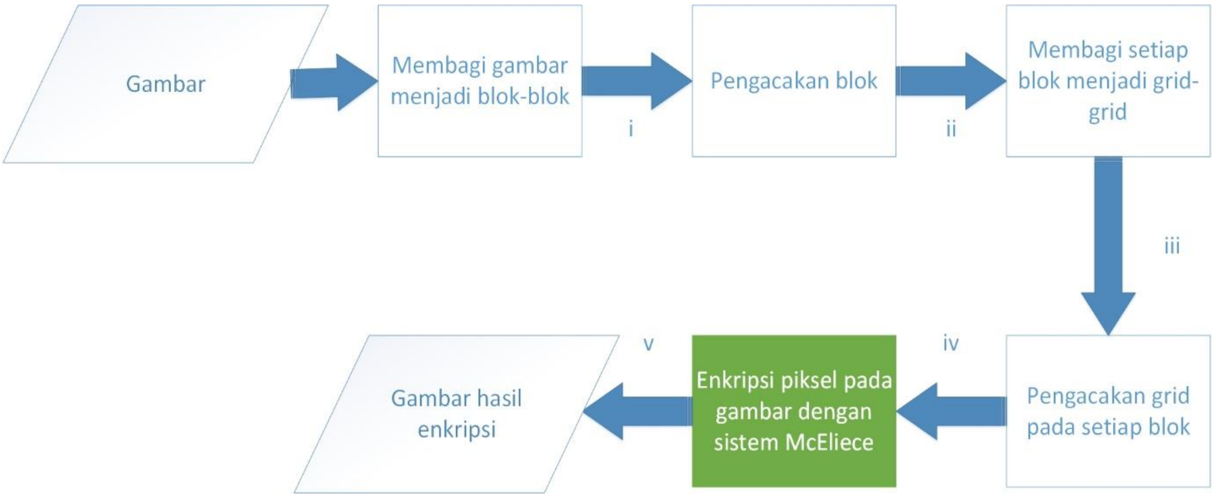
6. Dari m, untuk memperoleh pesan asli, nilai m diubah menjadi bilangan bulat, sehingga diperoleh pesan berupa piksel bernilai 14.

3. Sistem yang Dibangun

Pada bab ini, penulis mengajukan metode untuk enkripsi dan dekripsi gambar berbasis grid menggunakan McEliece Cryptography. Penjelasan pemodelan dibagi menjadi dua bagian, karena proses enkripsi dan dekripsi memiliki alur yang berbeda.

3.1. Enkripsi Gambar

Langkah-langkah yang dilakukan untuk enkripsi gambar antara lain membagi gambar menjadi beberapa blok, mengacak semua blok, membagi masing-masing blok menjadi beberapa grid, mengacak semua grid pada masing-masing blok, dan enkripsi setiap grid. Secara umum, sistem untuk enkripsi digambarkan dengan diagram dengan alur sebagai berikut.



Gambar 3.1 Alur Proses Enkripsi yang Dibangun

3.1.1. Pembagian Gambar dan Pengacakan Blok

Sebelum proses enkripsi dimulai, hal pertama yang harus dilakukan adalah membagi gambar menjadi beberapa blok. Kemudian, blok tersebut diacak dengan cara menukar nilai piksel pada blok satu dengan blok yang lain seperti algortima di bawah. Blok-blok pada gambar tersebut ditulis dengan persamaan matematika sebagai berikut.

$$frame = \sum_{l=0}^{l=N} \sum_{m=0}^{m=N} B_{lm}, \ldots \quad (1)$$

dimana l = baris blok (harus genap), m = kolom blok, B = blok pada gambar, N = jumlah baris dan kolom. Algoritma untuk pembagian gambar menjadi beberapa blok dan pengacakan blok ditulis sebagai berikut.

Algoritma 3 Pembagian Gambar dan Pengacakan blok

Input: Original Image

1: *Bagi gambar menjadi beberapa blok dengan ukuran l x m pada masing-masing blok*
2: *Memberikan indeks baris blok dan indeks kolom blok pada gambar, sesuai (1)*
3: $p \leftarrow 1$
4: $q \leftarrow l$
5: $while\ p \leq \frac{l}{2}\ and\ q \geq \frac{l}{2} + 1$
  $for\ j = 1\ to\ m\ do$
    $swap(B_{pm}, B_{qm})$

endfor
6: endwhile

Output: Shuffled-blocks Image

3.1.2. Pembagian Semua Blok dan Pengacakan Grid

Setelah gambar dibagi menjadi beberapa blok, setiap blok dibagi menjadi beberapa grid dan posisi semua grid di dalam setiap blok diacak dengan cara yang telah ditentukan. Grid-grid pada suatu blok ditulis dengan persamaan matematika sebagai berikut.

$$B_{11} = \sum_{n=0}^{n=M} \sum_{r=0}^{r=M} G_{nr}, \quad (2)$$

dimana n = baris grid, r = kolom grid, G = grid pada blok, M = jumlah baris dan kolom.

Algoritma untuk pembagian semua blok menjadi grid-grid dan pengacakan grid pada setiap blok ditulis sebagai berikut.

Algoritma 4 Pembagian Semua Blok dan Pengacakan Grid

Input: Shuffled-blocks Image

1: Pada masing-masing blok, bagi blok menjadi beberapa grid dengan ukuran n x r.
2: Memberikan indeks baris blok dan indeks kolom blok pada masing masing blok, sesuai (2)
3: For each block do
$p \leftarrow 1$
$q \leftarrow n$
while $p \leq \frac{n}{2}$ *and* $q \geq \frac{n}{2} + 1$
for $j = 1$ *to* r *do*
$swap(G_{pr}, G_{qr})$
endfor
$endwhile$
endfor

Output: Shuffled-grids Image

3.1.3. Enkripsi Grid

Setelah semua grid dalam setiap blok teracak, setiap grid dienkripsi dengan sistem kriptografi McEliece dan menghasilkan persamaan berikut.

$$G_{ij} \rightarrow G'_{ij}, \quad (3)$$

Algoritma untuk enkripsi setiap grid pada persamaan (iii) ditulis sebagai berikut.

Algoritma 5 Enkripsi grid

Input: Shuffled-grids Image

1: Tentukan nilai k dan n.
2: Pilih sembarang matriks A berdimensi k x $(n-k)$.
3: Pilih matriks generator berdimensi k x n dan $G = [I_k \mid A]$.
4: Pilih matriks non-singular S berdimensi k x n atau 4 x 4.
5: Pilih matriks permutasi P berdimensi n x n atau 7 x 7.
6: Simpan matriks S, G, dan P sebagai private key.
7: Hitung $G' = SGP$, yang merupakan public key.
8: Pilih sembarang vektor berdimensi 1 x n.
9: for each block do
10: for each grid in a block do
11: for each pixel in a grid do
Piksel gambar diubah menjadi bilangan biner.

Proses enkripsi untuk memperoleh ciphertext c dengan cara $c = m.G' \oplus e$.
end for
end for
end for
12: *Satukan kembali grid menjadi blok*
13: *Satukan kembali blok menjadi gambar*

Output: Encrypted Image

Ket:

$G_{ij} = pixel\ sebelum\ enkripsi, G'_{ij} = pixel\ setelah\ enkripsi.$

3.2. Dekripsi Gambar

Langkah-langkah yang dilakukan untuk dekripsi gambar adalah kebalikan dari proses enkripsi, antara lain membagi gambar terenkripsi menjadi beberapa grid, dekripsi semua grid, menyusun kembali urutan grid menjadi grid awal, menyatukan semua grid menjadi beberapa blok, menyusun kembali urutan blok ke blok awal, dan menyatukan semua blok menjadi gambar asli. Secara umum, sistem untuk dekripsi digambarkan dengan diagram alur sebagai berikut.

Gambar 3.2 Alur Proses Dekripsi yang Dibangun

3.2.1. Dekripsi Grid

Pada proses dekripsi, setiap grid didekripsi menjadi grid semula.

$$G'_{ij} \rightarrow G_{ij}, \quad (4)$$

Algoritma untuk dekripsi setiap grid pada persamaan (4) ditulis sebagai berikut.

Algoritma 6 Dekripsi Gambar

Input: Encrypted Image

Divide image into P blocks;
Divide image into Q grids;
Menghitung (P^{-1}), invers dari P;
for each block do
for each grid in a block do
for each pixel in a grid do
Menghitung $\hat{c} = c(P^{-1})$. // c = *ciphertext* dari piksel gambar hasil enkripsi
Proses *decoding* $\hat{c}$ menjadi $\hat{m}$.
Kemudian dari $\hat{m}$, diambil k digit pertamanya.
Menghitung $m = \hat{m}S^{-1}$.
Nilai m diubah menjadi bilangan bulat.
end for
end for

end for
5: Satukan kembali grid menjadi blok
6: Satukan kembali blok menjadi gambar

Output: Decrypted-grids Image

Ket:

$G'_{ij} = pixel\ sebelum\ dekripsi, G_{ij} = pixel\ setelah\ dekripsi.$

3.1.1. Penyusunan Kembali Setiap Grid dan Penyatuan menjadi Blok

Setelah setiap grid didekripsi, semua grid pada setiap blok dikembalikan pada urutan semula dan ditulis dengan persamaan sebagai berikut.

$$B_{11} = \sum_{n=0}^{n=M} \sum_{r=0}^{r=M} G_{nr}, \quad (5)$$

dimana n = baris grid, r = kolom grid, G = grid yang telah didekripsi pada blok, M = jumlah baris dan kolom.

Algoritma untuk penyusunan kembali setiap grid dan penyatuan menjadi blok ditulis sebagai berikut.

Algoritma 7 Penyusunan Kembali Setiap Grid dan Penyatuan menjadi Blok

Input: Decrypted-grids Image

1: Pada masing-masing blok, bagi blok menjadi beberapa grid dengan ukuran $n\ x\ r$.
2: Memberikan indeks baris blok dan indeks kolom blok pada masing masing blok, sesuai (5)
3: For each block do
$p \leftarrow 1$
$q \leftarrow n$
while $p \leq \frac{n}{2}$ *and* $q \geq \frac{n}{2} + 1$
for $j = 1$ *to* r *do*
$swap(G_{pr}, G_{qr})$
endfor
endwhile
endfor

Output: Combined-grids Image

3.1.2. Penyusunan Kembali Setiap Blok dan Penyatuan menjadi Gambar Asli

Setelah itu, setiap blok pada gambar dikembalikan pada urutan semula hingga kembali menjadi gambar asli dan ditulis dengan persamaan sebagai berikut.

$$frame = \sum_{l=0}^{l=N} \sum_{m=0}^{m=N} B_{lm}, \quad (6)$$

dimana l = baris blok, m = kolom blok, B = blok yang didekripsi pada gambar, N = jumlah baris dan kolom.

Algoritma untuk penyusunan kembali setiap blok dan penyatuan menjadi gambar asli ditulis sebagai berikut.

Algoritma 8 Penyusunan Kembali Setiap Blok dan Penyatuan menjadi Gambar Asli

Input: Combined-grids Image

1: Bagi gambar menjadi beberapa blok dengan ukuran $l\ x\ m$ pada masing-masing blok
2: Memberikan indeks baris blok dan indeks kolom blok pada gambar, sesuai (6)
3: $p \leftarrow 1$
4: $q \leftarrow l$

5: while $p \leq \frac{l}{2}$ *and* $q \geq \frac{l}{2} + 1$
for $j = 1$ *to* m *do*
$swap(B_{pm}, B_{qm})$
endfor
6: endwhile

Output: Original Image

4. Evaluasi

Pada bagian ini akan dipaparkan hasil enkripsi dan dekripsi gambar, analisis gambar hasil enkripsi berdasarkan histogram, dan analisis terhadap perbandingan waktu komputasi pada algoritma RSA dan algoritma McEliece.

4.1. Spesifikasi Sistem

Pada bagian ini, akan dijelaskan spesifikasi system, spesifikasi RSA dan spesifikasi McEliece. Spesifikasi sistem yang digunakan untuk proses evaluasi adalah sebagai berikut.

1. System: Windows 7 Ultimate 64-bit Operating System
2. Processor: AMD A6-1450 @ 1.4 GHz
3. RAM: 4 GB

Untuk spesifikasi RSA, kunci yang dipilih antara lain $p = 37$, $q = 43$, $n = 1591$, yang merupakan hasil perkalian dari p dan q, $\phi(n) = 1512$, $e = 53$, dan $d = 485$, yang ditentukan berdasarkan $1 < d < \phi(n)$. Pada algoritma McEliece, spesifikasi yang dipilih untuk membangkitkan kunci antara lain $n = 32$, $k = 12$, $m = 5$, dan $t = 4$.

4.2. Pengujian Enkripsi dengan Algoritma RSA dan Algoritma McEliece

Pengujian dilakukan terhadap algoritma RSA dan algoritma McEliece untuk menguji tingkat keacakan gambar melalui histogram gambar hasil enkripsi Histogram suatu gambar menggambarkan nilai intensitas piksel pada gambar berdasarkan nilai entropi pada gambar. Entropi adalah ukuran statistik keacakan yang dapat digunakan untuk mengkarakterisasi tekstur gambar input. Nilai entropi masing-masing piksel merah, piksel hijau, dan piksel biru dihitung dengan rumus berikut [9].

$$E_r = \sum_{i=1}^{n} p_r(x_i) \,.\, (\log_2(p_r(x_i))) \quad \text{.....(7)}$$

$$E_g = \sum_{i=1}^{n} p_g(x_i) \,.\, (\log_2(p_g(x_i))) \quad \text{.....(8)}$$

$$E_b = \sum_{i=1}^{n} p_b(x_i) \,.\, (\log_2(p_b(x_i))) \quad \text{.....(9)}$$

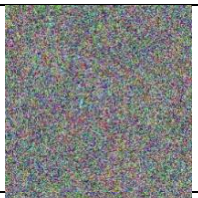
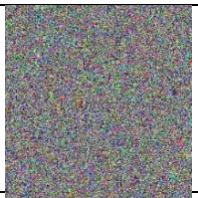
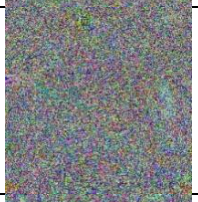
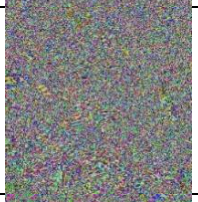
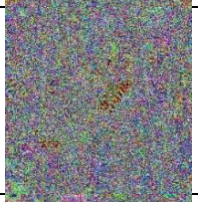
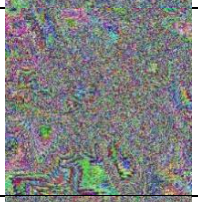
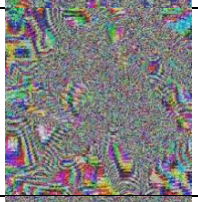
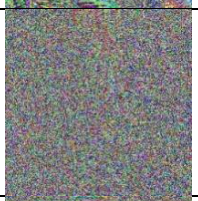
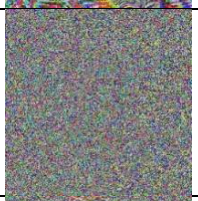
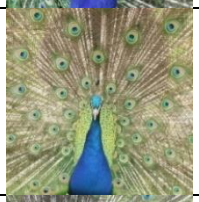
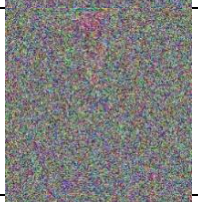
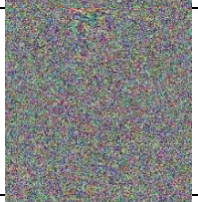
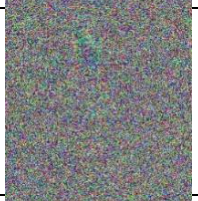
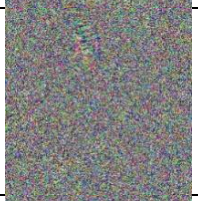
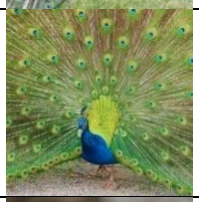
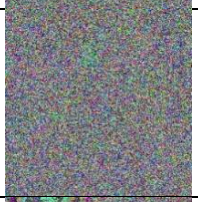
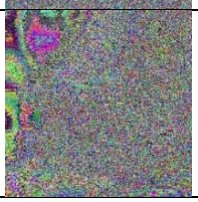
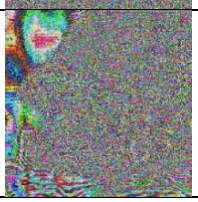
Setelah dihitung semua entropi, kemudian dihitung rata-rata dari 3 nilai entropi di atas, yaitu:

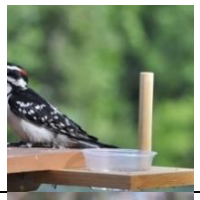
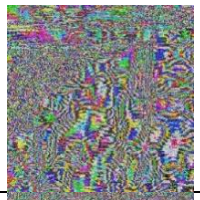
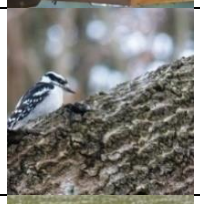
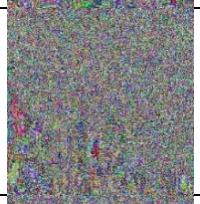
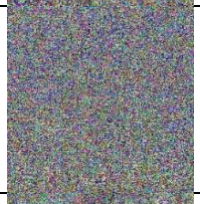
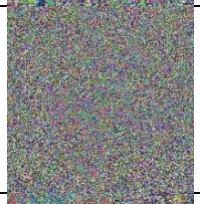
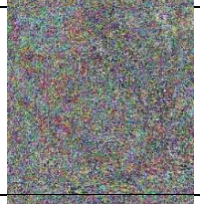
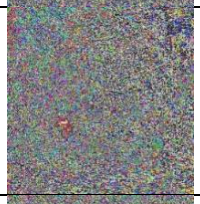
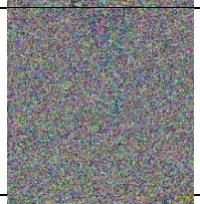
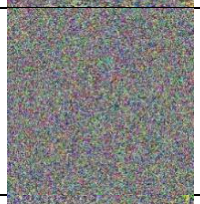
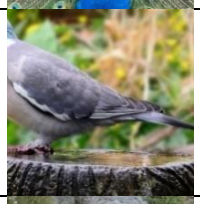
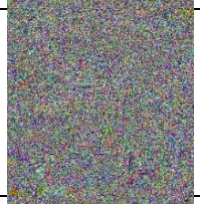
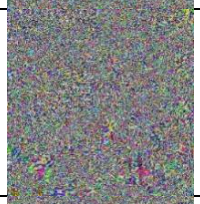
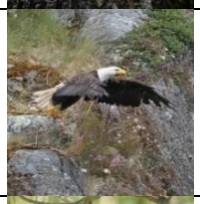
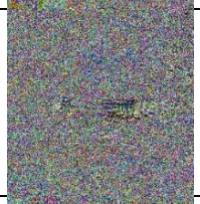
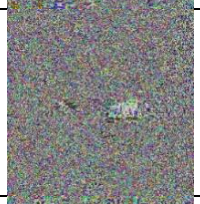
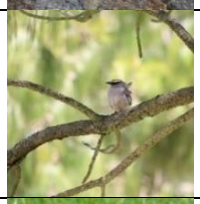
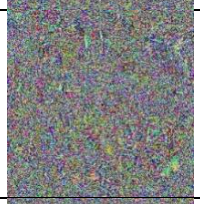
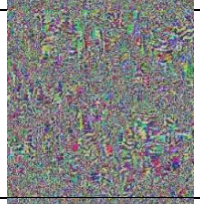
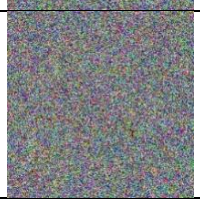
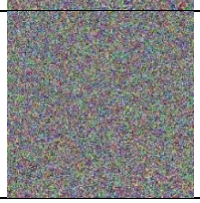
$$E = \frac{E_r + E_g + E_b}{3} \quad \text{.....(10)}$$

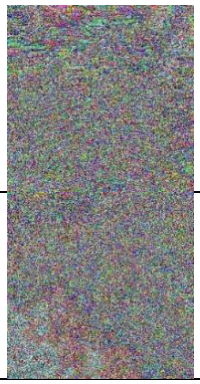
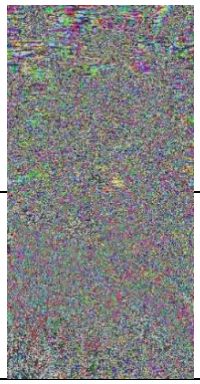
Semakin tinggi nilai entropi, semakin acak persebaran piksel pada gambar. Hasil dapat dilihat pada tabel-tabel berikut berupa table nilai entropi gambar dan tabel presentase perbedaan nilai entropi hasil enkripsi dengan gambar asli.

Tabel 4.1 Hasil Enkripsi dan Perbandingan Tingkat Keacakan Piksel untuk Gambar Histogram Tengah

	Gambar Asli			Entropi Gambar

No		Hasil Enkripsi RSA	Hasil Enkripsi McEliece	Gambar Asli	RSA	McEliece
1				7.64436498	6.992698711	6.905375234
2				7.551566327	6.937115451	6.805272816
3				7.27538827	6.764744639	6.66847186
4				7.381428779	6.837089105	6.755982058
5				7.551970123	6.921901759	6.911816568
6				7.253201194	6.721784346	6.743062099
7				7.318910121	6.774633051	6.831231914
8				7.108748379	6.658964371	6.675564464
9				7.3669854	6.807393112	6.77987822

10				7.305920175	6.770338172	6.714726345
11				7.678635621	7.016268859	6.898307185
12				7.069199402	6.608930555	6.592597239
13				6.585198878	6.282761033	6.346825087
14				7.665290957	7.001106791	6.941867511
15				7.781103366	7.107205809	6.934435954
16				7.176629188	6.71160623	6.700580399
17				7.36877934	6.780339446	6.738325443
18				7.175219358	6.703869996	6.623581152

19				7.57957628	6.947557309	6.930885083
20				7.357218027	6.806156121	6.90790269
Rata-rata Entropi				7.359766708	6.807623243	6.770334466

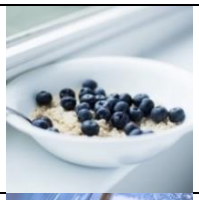
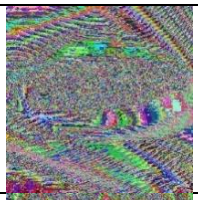
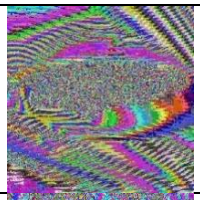
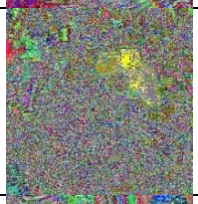
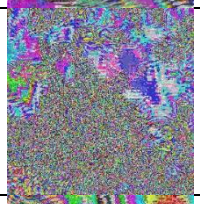
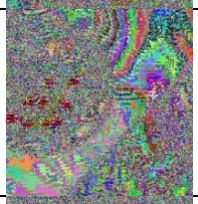
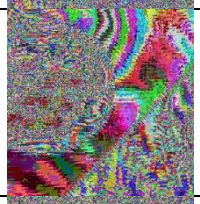
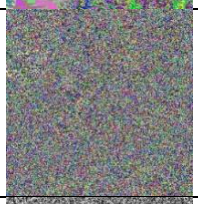
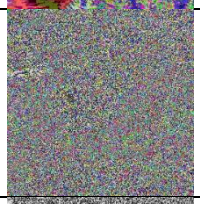
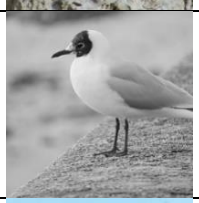
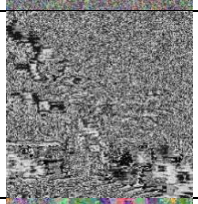
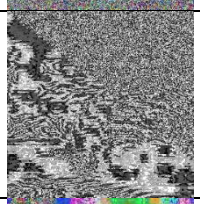
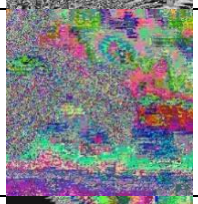
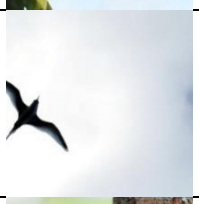
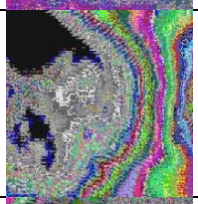
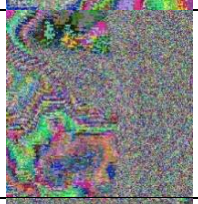
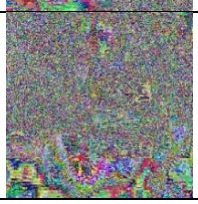
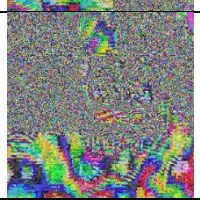
Berdasarkan tabel 4.1 di atas, hasil enkripsi RSA memliki nilai entropi sedikit lebih besar daripada hasil enkripsi McEliece jika ditinjau dari rata-rata nilai entropi pada 20 gambar di atas. Sehingga dapat disimpulkan bahwa, gambar hasil enkripsi RSA memiliki tingkat keacakan piksel lebih besar daripada gambar hasil enkripsi McEliece.

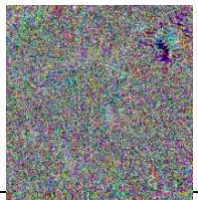
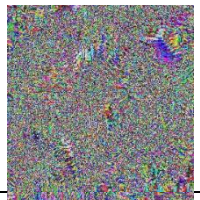
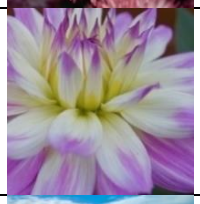
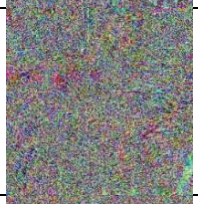
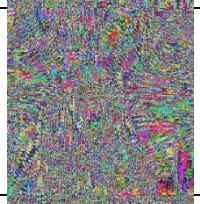
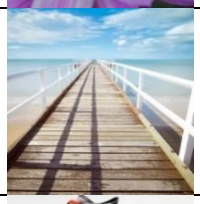
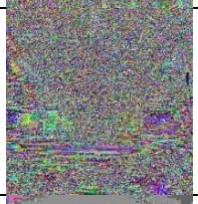
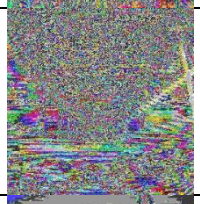
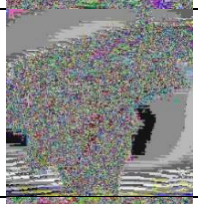
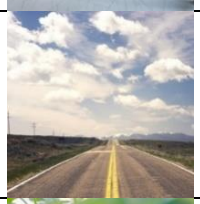
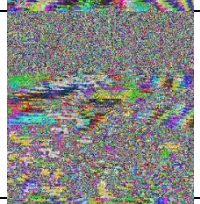
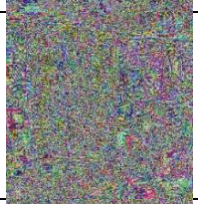
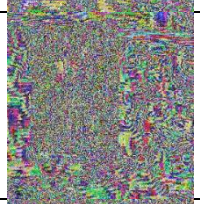
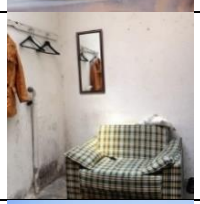
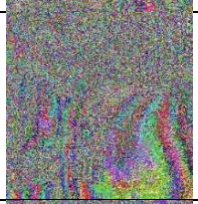
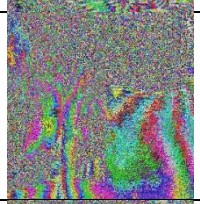
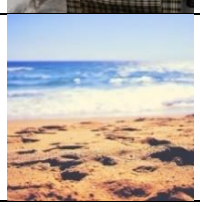
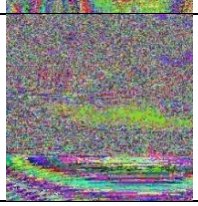
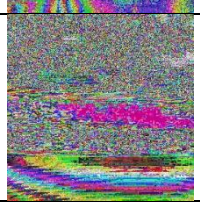
Tabel 4.2 Persentase Perbandingan Tingkat Keacakan Piksel untuk Gambar Histogram Tengah

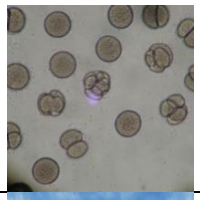
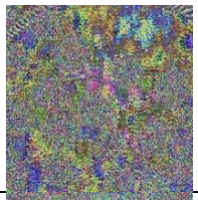
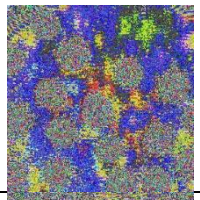
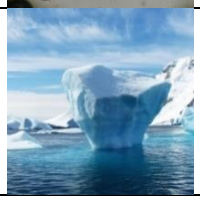
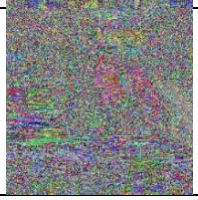
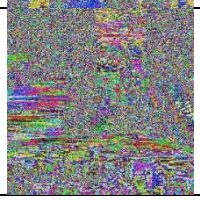
Gambar ke-	Entropi Gambar			Persentase Perbedaan dengan Gambar Asli		Persentase Perbedaan antara RSA dan McEliece
	Gambar Asli	RSA	McEliece	RSA	McEliece	
1	7.64436498	6.992698711	6.905375234	8.524792712	9.667117506	14.94335759
2	7.551566327	6.937115451	6.805272816	8.136734147	9.882632011	23.11967859
3	7.27538827	6.764744639	6.66847186	7.018781841	8.34204838	18.18826006
4	7.381428779	6.837089105	6.755982058	7.374448628	8.473247386	14.8859901
5	7.551970123	6.921901759	6.911816568	8.343099269	8.476643109	1.768331141
6	7.253201194	6.721784346	6.743062099	7.326652523	7.033295802	4.04451377
7	7.318910121	6.774633051	6.831231914	7.436586327	6.663262684	10.56610384
8	7.108748379	6.658964371	6.675564464	6.327189891	6.093673496	3.284915752
9	7.3669854	6.807393112	6.77987822	7.595946743	7.969435914	5.06976939
10	7.305920175	6.770338172	6.714726345	7.330794612	8.091983152	10.41879081
11	7.678635621	7.016268859	6.898307185	8.626099666	10.16233188	20.00657785
12	7.069199402	6.608930555	6.592597239	6.510904858	6.741953875	3.268390147
13	6.585198878	6.282761033	6.346825087	4.592691127	3.619841938	14.77326968
14	7.665290957	7.001106791	6.941867511	8.664826555	9.437651489	10.08213437
15	7.781103366	7.107205809	6.934435954	8.660694055	10.88107138	28.53550737
16	7.176629188	6.71160623	6.700580399	6.479684911	6.633320135	2.140771384
17	7.36877934	6.780339446	6.738325443	7.985581695	8.555744012	7.73754092
18	7.175219358	6.703869996	6.623581152	6.569128241	7.68810232	15.59498077
19	7.57957628	6.947557309	6.930885083	8.338447263	8.558409773	2.902042309
20	7.357218027	6.806156121	6.90790269	7.490085296	6.107136351	18.79717224
Rata-rata Persentase	7.359766708	6.807623243	6.770334466	7.466658518	7.95394513	11.5064049

Berdasarkan tabel persentase di atas, persentase perbedaan nilai entropi dari kedua algoritma tidak terlalu besar dengan gambar asli. Tetapi, algoritma McEliece memiliki persentase lebih daripad algoritma RSA berdasarkan rata-rata untuk gambar gambar histogram tengah.

Tabel 4.3 Hasil Enkripsi dan Perbandingan Tingkat Keacakan Piksel untuk Gambar Histogram Kiri

No	Gambar Asli	Hasil Enkripsi RSA	Hasil Enkripsi McEliece	Entropi Gambar		
				Gambar Asli	RSA	McEliece
1				7.06908124	6.571018182	6.434400633
2				7.090655492	6.532155591	6.379658216
3				6.963724214	6.504264922	6.250288146
4				7.315410851	6.757434819	6.774831456
5				6.912268395	6.450444627	6.478110811
6				5.971824915	5.626947954	5.671813113
7				5.595620652	5.369869468	5.488327444
8				7.3179427	6.728630388	6.733407608
9				7.263445619	6.694350274	6.725452852

10				7.6870716	7.013392385	6.802817246
11				6.929529454	6.491667442	6.631242359
12				7.479809712	6.820619844	6.656336488
13				6.901485466	6.38060387	6.223491805
14				7.428146964	6.797704138	6.729193702
15				7.467685561	6.819020666	6.720056074
16				7.528240748	6.938510271	6.871474478
17				7.555888212	6.890895314	6.782490348
18				7.127997426	6.575816909	6.459636313

19				5.869725339	5.646514288	5.765312028
20				7.60436557	6.934024325	6.770457437
Rata-rata Entropi				7.053996006	6.527194284	6.467439928

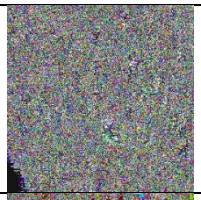
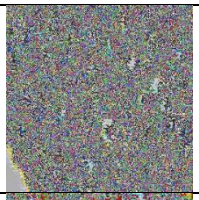
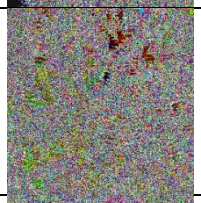
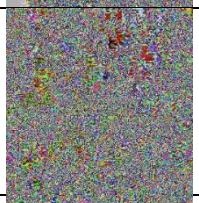
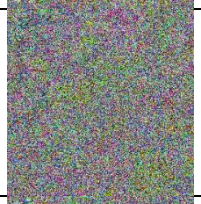
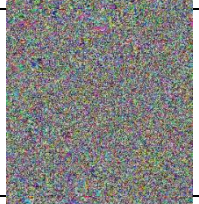
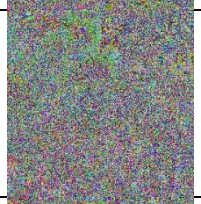
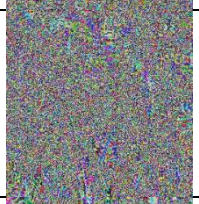
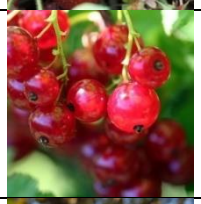
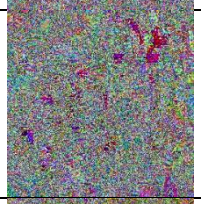
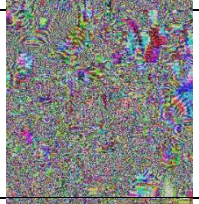
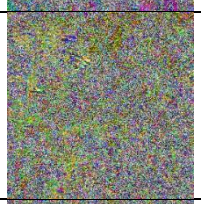
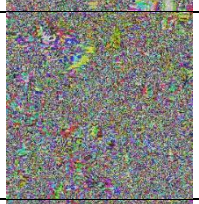
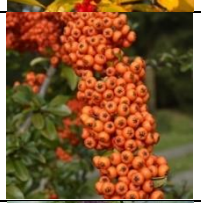
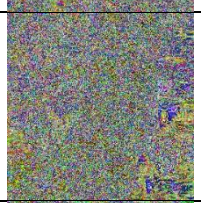
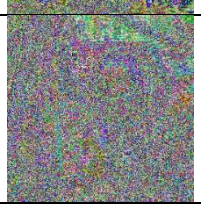
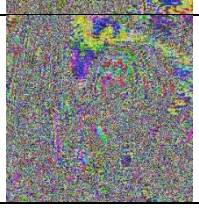
Berdasarkan tabel 4.2 di atas, hasil enkripsi RSA memliki nilai entropi sedikit lebih besar daripada hasil enkripsi McEliece jika ditinjau dari rata-rata nilai entropi pada 20 gambar di atas. Sehingga dapat disimpulkan bahwa, gambar hasil enkripsi RSA memiliki tingkat keacakan piksel lebih besar daripada gambar hasil enkripsi McEliece.

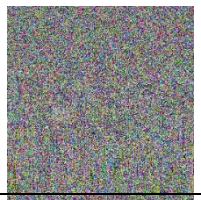
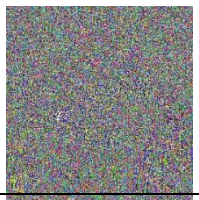
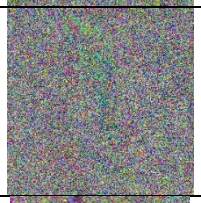
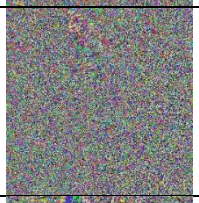
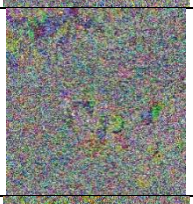
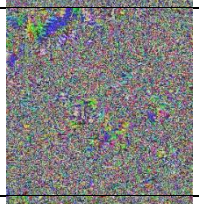
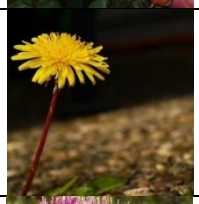
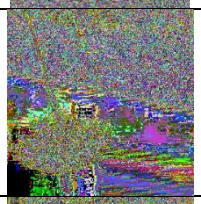
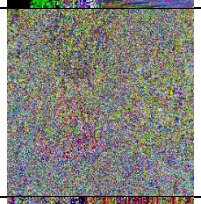
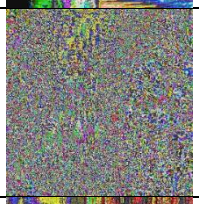
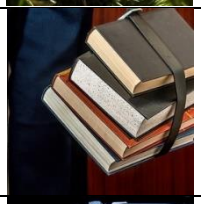
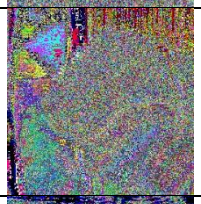
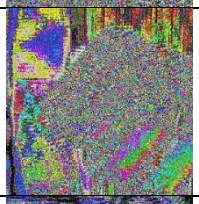
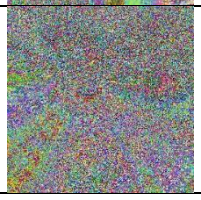
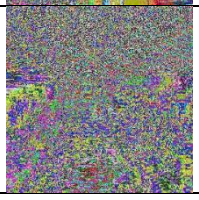
Tabel 4.4 Persentase Perbandingan Tingkat Keacakan Piksel untuk Gambar Histogram Kiri

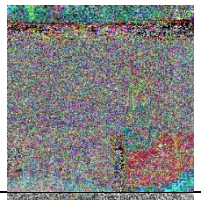
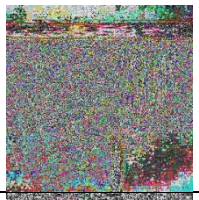
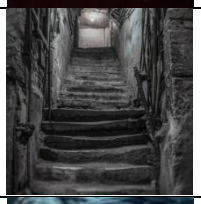
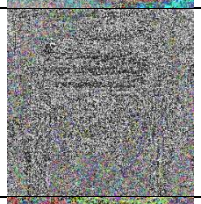
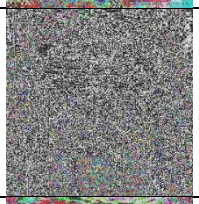
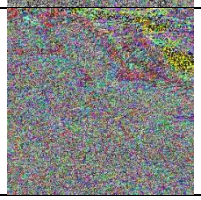
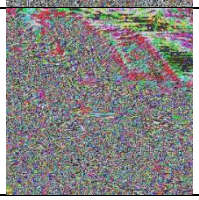
Gambar ke-	Entropi Gambar			Persentase Perbedaan		Persentase Perbedaan antara RSA dan McEliece
	Gambar Asli	RSA	McEliece	RSA	McEliece	
1	7.06908124	6.571018182	6.434400633	7.045654739	8.978261611	27.33886917
2	7.090655492	6.532155591	6.379658216	7.876562353	10.02724327	30.33120025
3	6.963724214	6.504264922	6.250288146	6.597896149	10.24503622	52.37341335
4	7.315410851	6.757434819	6.774831456	7.62740526	7.389597197	3.250782058
5	6.912268395	6.450444627	6.478110811	6.681218688	6.280971151	5.790393455
6	5.971824915	5.626947954	5.671813113	5.775068196	5.023787644	12.58041825
7	5.595620652	5.369869468	5.488327444	4.034426164	1.917449631	37.83273859
8	7.3179427	6.728630388	6.733407608	8.052977957	7.98769704	0.892066518
9	7.263445619	6.694350274	6.725452852	7.835060306	7.406853366	5.895369265
10	7.6870716	7.013392385	6.802817246	8.763795242	11.50313669	35.63569581
11	6.929529454	6.491667442	6.631242359	6.318784203	4.304579377	29.06697836
12	7.479809712	6.820619844	6.656336488	8.812922972	11.0092804	29.36381427
13	6.901485466	6.38060387	6.223491805	7.547383795	9.823880152	32.98559953
14	7.428146964	6.797704138	6.729193702	8.487215297	9.409523869	12.41640178
15	7.467685561	6.819020666	6.720056074	8.686290943	10.01152875	17.74629899
16	7.528240748	6.938510271	6.871474478	7.833576224	8.724033839	11.8282298
17	7.555888212	6.890895314	6.782490348	8.800989104	10.23569754	18.98795215
18	7.127997426	6.575816909	6.459636313	7.746643038	9.376562204	22.86643876
19	5.869725339	5.646514288	5.765312028	3.802751209	1.778844915	34.48042587
20	7.60436557	6.934024325	6.770457437	8.815215926	10.96617627	28.28586193
Rata-rata Persentase	7.053996006	6.527194284	6.467439928	7.357091888	8.120007057	22.49744741

Berdasarkan tabel persentase di atas, persentase perbedaan nilai entropi dari kedua algoritma tidak terlalu besar dengan gambar asli. Tetapi, algoritma McEliece memiliki persentase lebih daripada algoritma RSA berdasarkan rata-rata untuk gambar gambar histogram kiri.

Tabel 4.5 Hasil Enkripsi dan Perbandingan Tingkat Keacakan Piksel untuk Gambar Histogram Kanan

No	Gambar Asli	Hasil Enkripsi RSA	Hasil Enkripsi McEliece	Entropi Gambar		
				Gambar Asli	RSA	McEliece
1				7.516969046	7.002281066	6.676474385
2				7.221489069	6.78062113	6.445369225
3				7.304146411	6.817993793	6.55238752
4				7.432654764	6.90538018	6.638167347
5				7.494395348	6.948468138	6.671413055
6				7.074081128	6.641021008	6.392747993
7				7.115550896	6.681214177	6.35603668
8				7.491624883	6.925500476	6.679719237

9				7.796464026	7.137312464	6.87137374
10				7.299689812	6.82017985	6.50035668
11				7.4661987	6.951528458	6.716827133
12				6.724852786	6.400141802	6.068765514
13				7.098429416	6.698918468	6.363094408
14				7.248994612	6.776879074	6.468955456
15				5.133779488	4.929355131	4.815214369
16				6.937170086	6.593766695	6.275732563
17				7.021467164	6.627383984	6.325759469

18				7.376410107	6.881175678	6.529585668
19				7.069227044	6.691077827	6.302254803
20				7.21221212	6.782402625	6.428157417
Rata-rata Entropi				7.151790345	6.699630101	6.403919633

Berdasarkan tabel 4.3 di atas, hasil enkripsi RSA memliki nilai entropi sedikit lebih besar daripada hasil enkripsi McEliece jika ditinjau dari rata-rata nilai entropi pada 20 gambar di atas. Sehingga dapat disimpulkan bahwa, gambar hasil enkripsi RSA memiliki tingkat keacakan piksel lebih besar daripada gambar hasil enkripsi McEliece.

Tabel 4.6 Persentase Perbandingan Tingkat Keacakan Piksel untuk Gambar Histogram Kanan

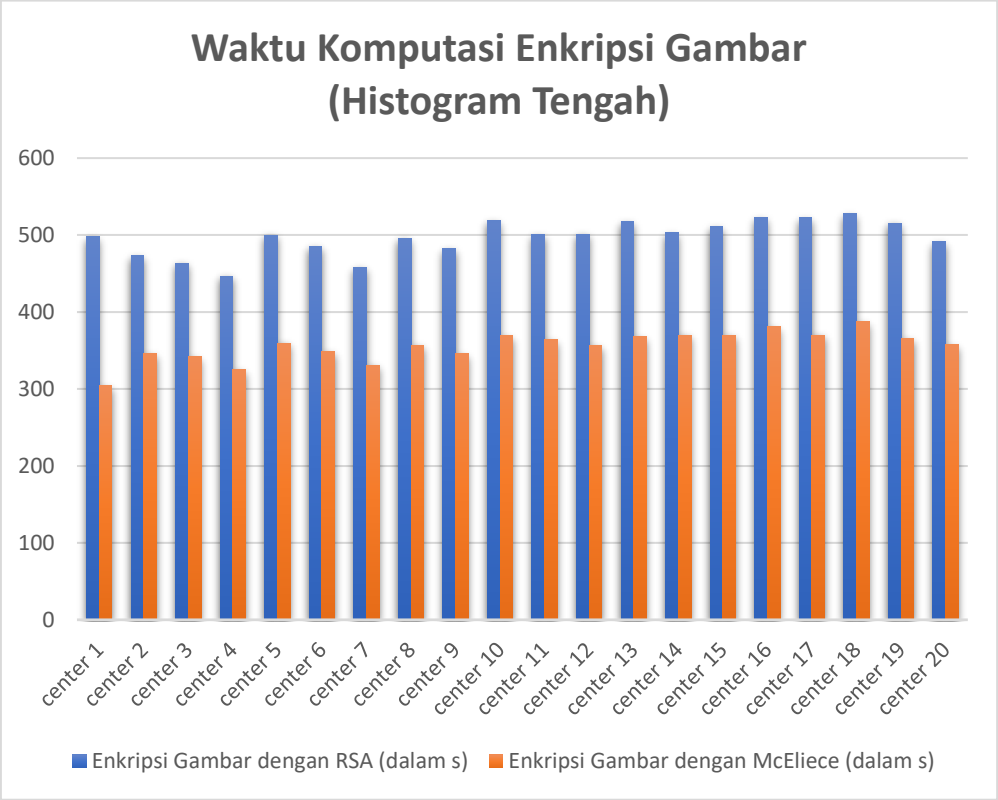
Gambar ke-	Entropi Gambar			Persentase Perbedaan		Persentase Perbedaan antara RSA dan McEliece
	Gambar Asli	RSA	McEliece	RSA	McEliece	
1	7.516969046	7.002281066	6.676474385	6.847014759	11.18129736	57.65997665
2	7.221489069	6.78062113	6.445369225	6.104945038	10.74736577	64.28619761
3	7.304146411	6.817993793	6.55238752	6.655844374	10.29222101	49.78510055
4	7.432654764	6.90538018	6.638167347	7.094027655	10.68914731	48.36925394
5	7.494395348	6.948468138	6.671413055	7.284473059	10.98130343	49.32793373
6	7.074081128	6.641021008	6.392747993	6.121786166	9.631401205	49.61230972
7	7.115550896	6.681214177	6.35603668	6.10404909	10.67400441	64.22489815
8	7.491624883	6.925500476	6.679719237	7.556763925	10.83751067	43.79219193
9	7.796464026	7.137312464	6.87137374	8.454493728	11.86551086	43.75082244
10	7.299689812	6.82017985	6.50035668	6.568908742	10.95023423	60.02070771
11	7.4661987	6.951528458	6.716827133	6.893337062	10.03685539	42.10333074
12	6.724852786	6.400141802	6.068765514	4.828521817	9.756158139	73.27500659
13	7.098429416	6.698918468	6.363094408	5.62815975	10.35912262	66.64802305
14	7.248994612	6.776879074	6.468955456	6.512841611	10.760653	58.59862803
15	5.133779488	4.929355131	4.815214369	3.981946578	6.205274672	43.30782223
16	6.937170086	6.593766695	6.275732563	4.950194196	9.534687987	66.0859361
17	7.021467164	6.627383984	6.325759469	5.612547492	9.908295211	61.18020094
18	7.376410107	6.881175678	6.529585668	6.713759422	11.48017025	64.61694449
19	7.069227044	6.691077827	6.302254803	5.349230039	10.84944982	77.80510861
20	7.21221212	6.782402625	6.428157417	5.959468308	10.8712097	68.10311883
Rata-rata Persentase	7.151790345	6.699630101	6.403919633	6.261115641	10.38059365	57.6276756

Berdasarkan tabel persentase di atas, persentase perbedaan nilai entropi dari kedua algoritma tidak terlalu besar dengan gambar asli. Tetapi, algoritma McEliece memiliki persentase lebih daripada algoritma RSA berdasarkan rata-rata untuk gambar gambar histogram kanan.

Dari hasil analisis tabel-tabel di atas, dapat disimpulkan bahwa hasil enkripsi RSA memliki persentase perbedaan nilai entropi lebih kecil daripada hasil enkripsi McEliece, sehingga gambar hasil enkripsi RSA memiliki tingkat keacakan piksel lebih kecil daripada gambar hasil enkripsi McEliece. Hasil enkripsi pada histogram kanan memiliki persentase perbedaan nilai entropi paling besar dibandingkan dengan dua kelompok gambar lain.

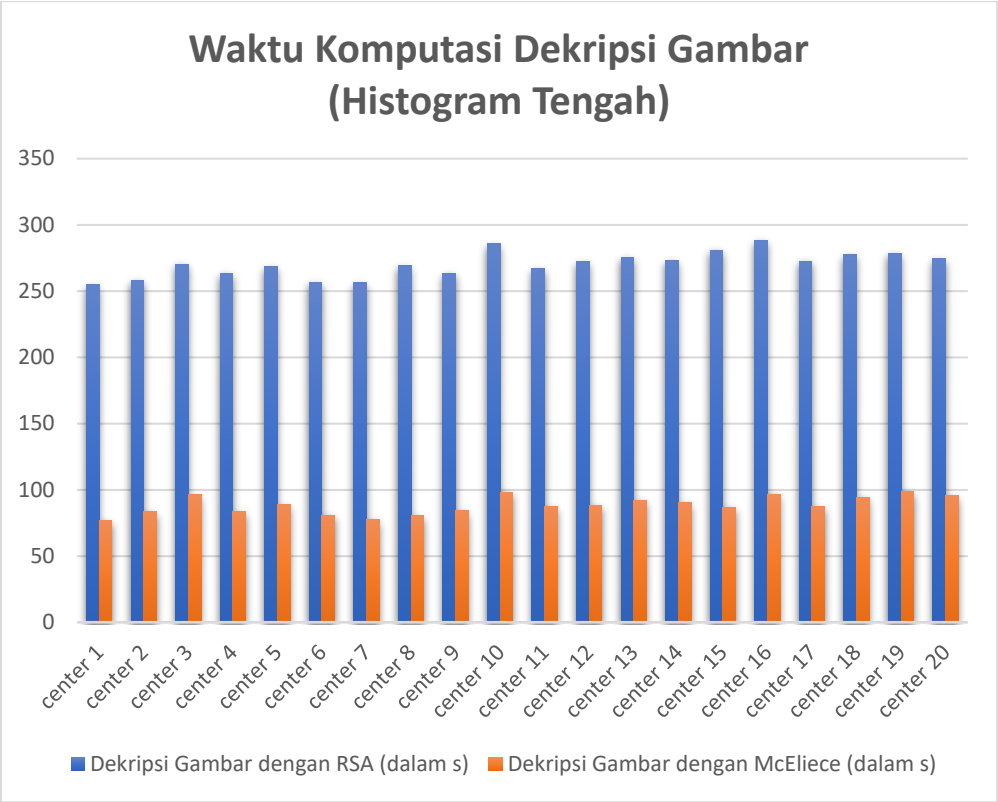
4.3. Analisis Waktu Komputasi

Pengujian dilakukan terhadap algoritma RSA dan algoritma McAliece untuk membandingkan waktu yang diperlukan untuk pengujian pada gambar *.png berukuran $256 \, x \, 256$. Hasil pengujian dapat dilihat pada grafik-grafik berikut. Grafik dibuat berdasarkan pada tabel hasil pengujian yang dapat dilihat pada lampiran.



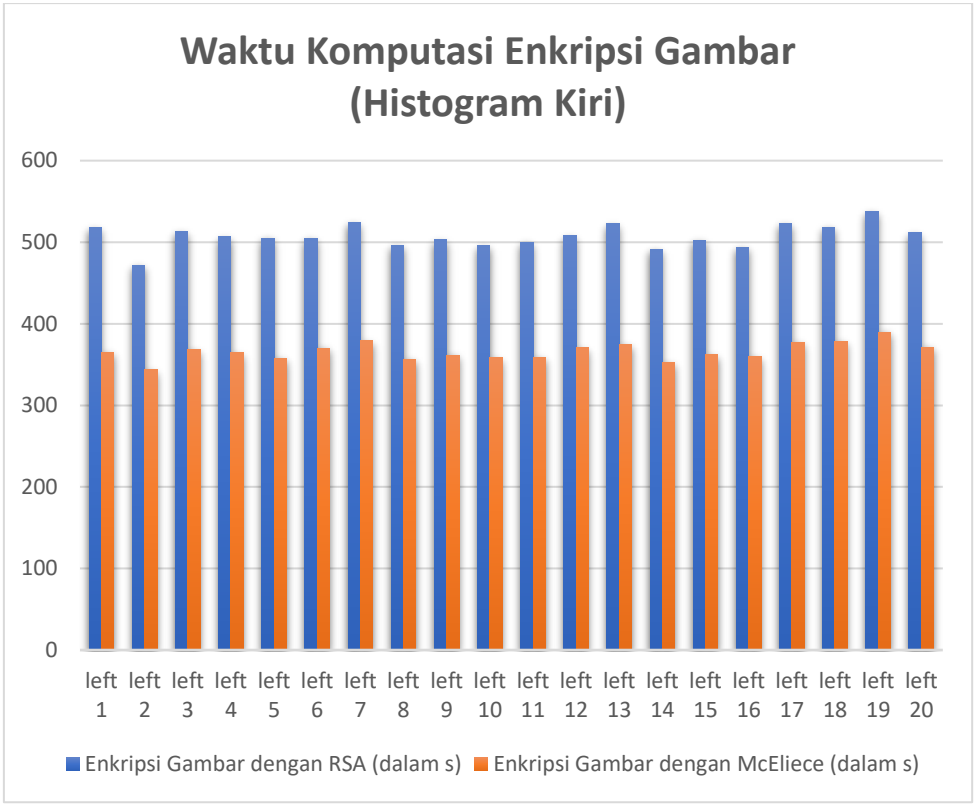
Gambar 4.1 Grafik Perbandingan Waktu Enkripsi untuk Gambar Histogram Tengah

Berdasarkan gambar 4.1 di atas, dapat diinformasikan bahwa waktu yang diperlukan untuk proses enkripsi gambar histogram tengah dengan sistem RSA memerlukan waktu yang lebih lama daripada proses enkripsi dengan sistem McEliece.



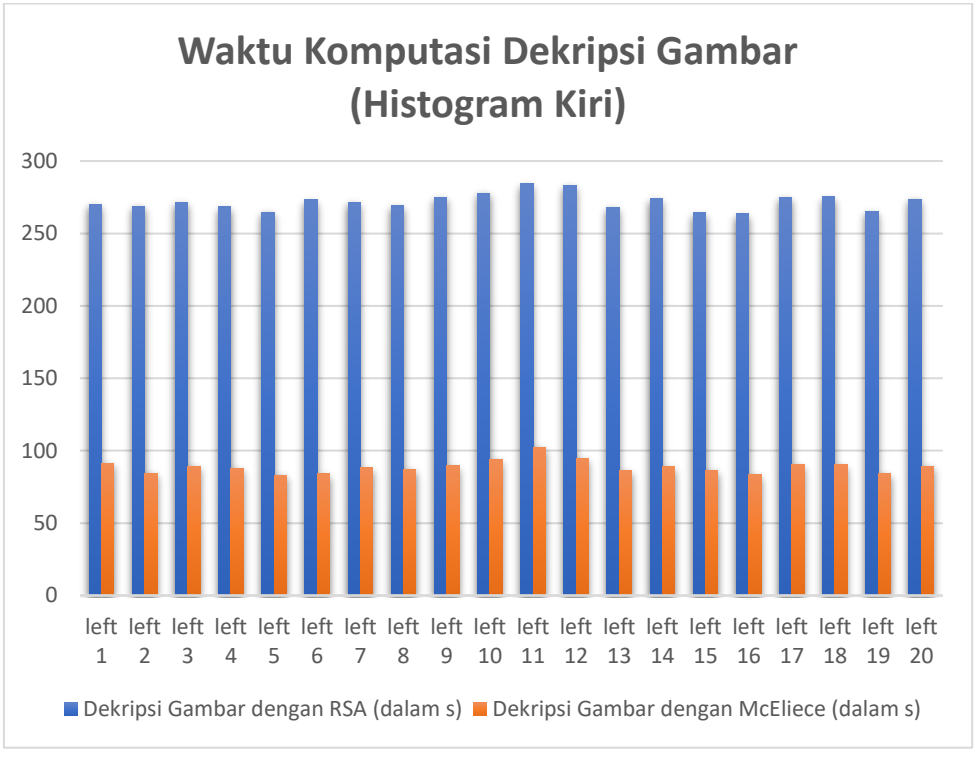
Gambar 4.2 Grafik Perbandingan Waktu Dekripsi untuk Gambar Histogram Tengah

Berdasarkan gambar 4.2 di atas, dapat diinformasikan bahwa waktu yang diperlukan untuk proses dekripsi gambar histogram tengah dengan sistem RSA memerlukan waktu yang lebih lama daripada proses dekripsi dengan sistem McEliece. Selain itu, sistem RSA dan sistem McEliece memerlukan waktu yang lebih lama untuk melakukan proses enkripsi gambar daripada melakukan proses dekripsi gambar, karena saat enkripsi, hasil enkripsi perlu dilakukan operasi modulo dengan 256 untuk memenuhi rentang nilai piksel gambar yaitu 0-255 dan nilai hasil bagi piksel enkripsi dengan 256 disimpan dalam suatu variabel untuk mengembalikan nilai enkrpsi sebenernya sebelum melakukan dekripsi. [7].



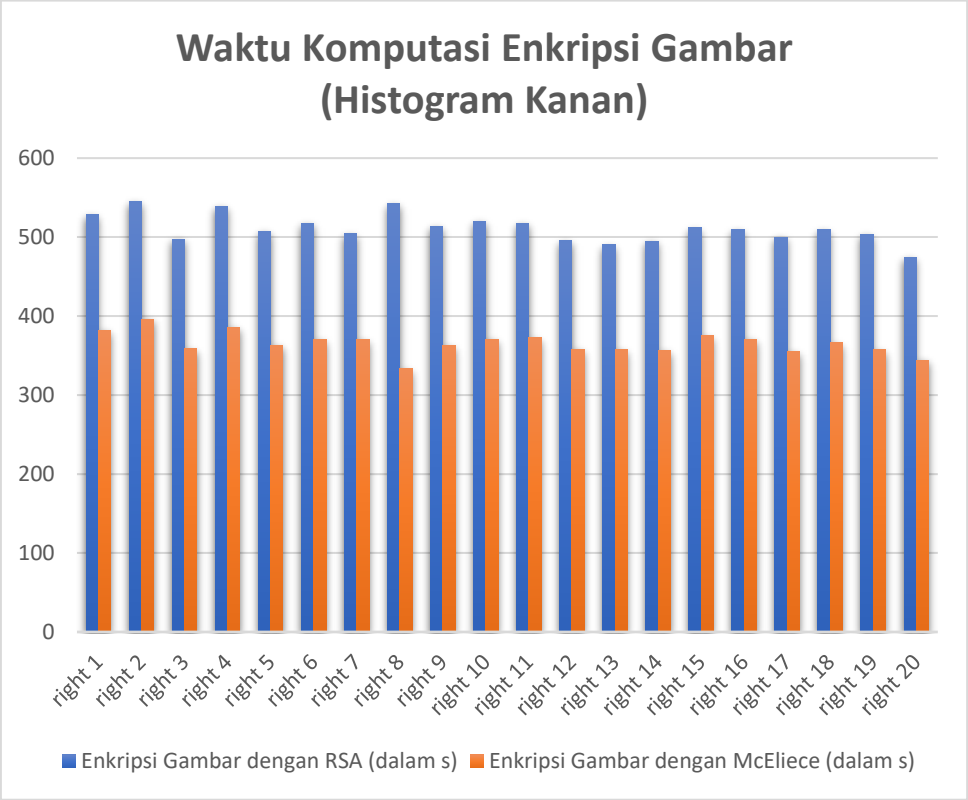
Gambar 4.3 Grafik Perbandingan Waktu Enkripsi untuk Gambar Histogram Kiri

Berdasarkan gambar 4.3 di atas, dapat diinformasikan bahwa waktu yang diperlukan untuk proses enkripsi gambar histogram kiri dengan sistem RSA memerlukan waktu yang lebih lama daripada proses enkripsi dengan sistem McEliece.



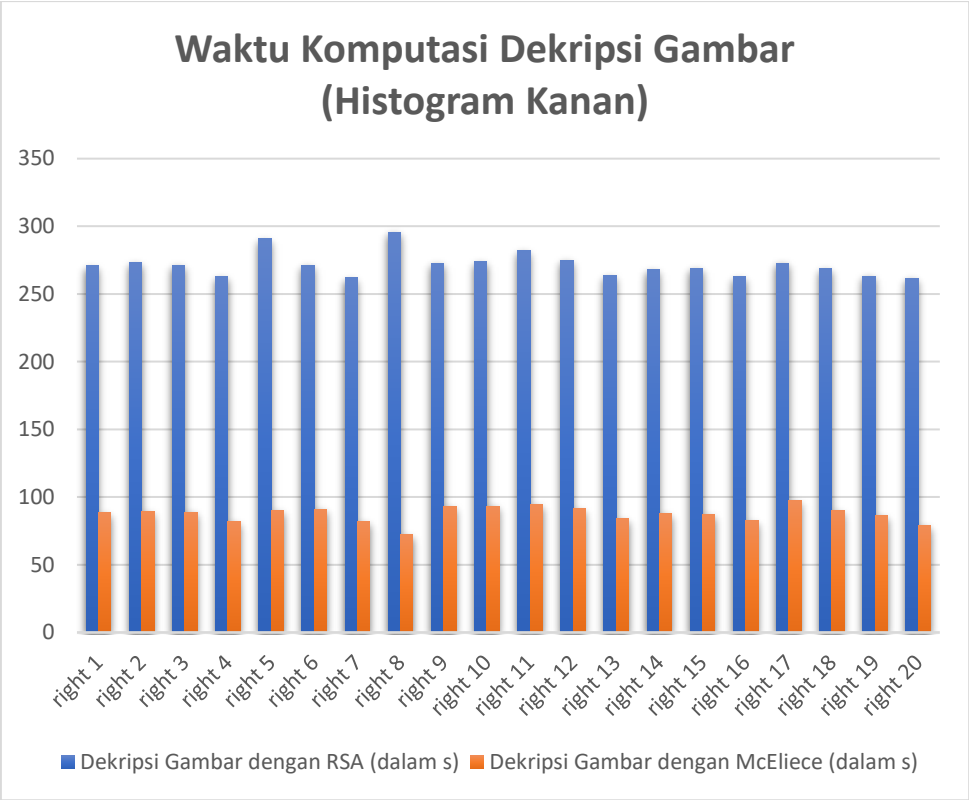
Gambar 4.4 Grafik Perbandingan Waktu Dekripsi untuk Gambar Histogram Kiri

Berdasarkan gambar 4.4 di atas, dapat diinformasikan bahwa waktu yang diperlukan untuk proses dekripsi gambar histogram kiri dengan sistem RSA memerlukan waktu yang lebih lama daripada proses dekripsi dengan sistem McEliece. Selain itu, sistem RSA dan sistem McEliece memerlukan waktu yang lebih lama untuk melakukan proses enkripsi gambar daripada melakukan proses dekripsi gambar, karena saat enkripsi, hasil enkripsi perlu dilakukan operasi modulo dengan 256 untuk memenuhi rentang nilai piksel gambar yaitu 0-255 dan nilai hasil bagi piksel enkripsi dengan 256 disimpan dalam suatu variabel untuk mengembalikan nilai enkrpsi sebenernya sebelum melakukan dekripsi. [7].



Gambar 4.5 Grafik Perbandingan Waktu Enkripsi untuk Gambar Histogram Kanan

Berdasarkan gambar 4.5 di atas, dapat diinformasikan bahwa waktu yang diperlukan untuk proses enkripsi gambar histogram kanan dengan sistem RSA memerlukan waktu yang lebih lama daripada proses enkripsi dengan sistem McEliece.



Gambar 4.6 Grafik Perbandingan Waktu Dekripsi untuk Gambar Histogram Kanan

Berdasarkan gambar 4.6 di atas, dapat diinformasikan bahwa waktu yang diperlukan untuk proses dekripsi gambar histogram kanan dengan sistem RSA memerlukan waktu yang lebih lama daripada proses dekripsi dengan sistem McEliece. Selain itu, sistem RSA dan sistem McEliece memerlukan waktu yang lebih lama untuk melakukan proses enkripsi gambar daripada melakukan proses dekripsi gambar, , karena saat enkripsi, hasil enkripsi perlu dilakukan operasi modulo dengan 256 untuk memenuhi rentang nilai piksel gambar yaitu 0-255 dan nilai hasil bagi piksel enkripsi dengan 256 disimpan dalam suatu variabel untuk mengembalikan nilai enkrpsi sebenernya sebelum melakukan dekripsi. [7].

Berdasarkan grafik-grafik di atas, dapat diinformasikan bahwa waktu yang diperlukan untuk proses enkripsi dan dekripsi dengan McEliece *Cryptosystem* lebih cepat dibandingkan RSA *Cryptosystem*. Kedua algortima memerlukan waktu yang lebih lama untuk melakukan proses enkripsi pesan daripada melakukan proses dekripsi gambar. Hal ini disebabkan oleh perpangkatan pada algoritma RSA untuk melakukan enkripsi dan dekripsi yang memiliki kompleksitas waktu $O((log\ n)\text{^}3)$ berdasarkan [10] dan lebih besar daripada kompleksitas perkalian matriks yang dilakukan untuk enkripsi dan kompleksitas waktu penghitungan invers matriks dalam proses dekripsi pada algoritma McEliece.

4.4. Analisis Kompleksitas Terhadap Waktu pada Algoritma RSA dan McEliece

Pada bagian ini, akan dipaparkan analisis kompleksitas waktu pada algoritma RSA dan algoritma McEliece.

4.4.1. Analisis Algoritma RSA

Sebelum melakukan enkripsi dan dekripsi, pembangkitan kunci dilakukan untuk menentukan kunci publik dan kunci privat. Analisis Algoritma Pembangkitan Kunci pada RSA adalah sebagai berikut.

a. Pilih dua bilangan prima berbeda p dan q yang besar agar tidak mudah difaktorkan.
b. Hitung $n = pq$. Setelah n diperoleh, bilangan tersebut akan digunakan sebagai kunci public
c. Hitung $\phi(n) = \phi(p).\phi(q) = (p-1)(q-1)$. ϕ adalah fungsi Euler.
d. Pilih bilangan bulat d sedemikian sehingga $1 < d < \phi(n)$ dan $gcd\ (d, \phi(n)) = 1$. d dan $\phi\ (n)$ adalah bilangan relatif prima. Setelah d diperoleh, angka tersebut akan digunakan sebagai kunci privat dari kedua pihak dan digunakan untuk dekripsi pesan.

e. Tentukan e dengan menghitung $e \equiv d^{-1} \ (mod\ \phi(n))$. e adalah invers perkalian modular dari d pada modulus $\phi(n)$, sehingga $d.e \equiv 1\ (mod\ \phi(n))$. Setelah itu, e digunakan sebagai kunci publik, sehingga (n, e) adalah kunci untuk enkripsi pesan.

Pada poin (a), algoritma yang digunakan untuk mengecek apakah bilangan yang dipilih bilangan prima atau tidak yaitu algoritma Miller-Rabin [8]. Kompleksitas dari algoritma Miller-Rabin adalah $O((log\ n)^3)$ [9]. Poin (b) dan (c) hanya dilakukan perkalian biasa, sehingga kompleksitasnya adalah $O((log\ n)^2)$. Poin (d) diperlukan algoritma rekursif *Euclid* yang memiliki kompleksitas $O((log\ n)^2)$. Poin (e) menggunakan algoritma rekursif *Extended Euclid* dengan kompleksitas waktu $O((log\ n)^2)$. Sehingga kompleksitas total diperoleh $O((log\ n)^3)$.

Setelah dilakukan pembangkitan kunci, proses enkripsi dan dekripsi dapat dilakukan. Enkripsi dilakukan dengan cara

$$\boldsymbol{C \equiv E(M)}$$
$$\boldsymbol{C \equiv M^e \ (mod\ n)}$$

dimana $\boldsymbol{C}$ adalah *Ciphertext* (Pesan yang telah dienkripsi), $\boldsymbol{E(M)}$ adalah fungsi enkripsi, $\boldsymbol{e}$ adalah kunci publik untuk enkripsi pesan, dan $\boldsymbol{M}$ adalah pesan asli. Karena proses enkripsi menggunakan algoritma perpangkatan modulo (*Modular Exponentiation*), sehingga kompleksitasnya adalah $O((log\ n)^3)$ [9].

Setelah C diterima, penerima melakukan proses dekripsi menggunakan kunci privat d yang telah diperoleh dari pembangkitan kunci. Dekripsi dilakukan dengan cara

$$\boldsymbol{M \equiv D(C)}$$
$$\boldsymbol{M \equiv C^d \ (mod\ n)}$$

dimana $\boldsymbol{M}$ adalah pesan asli, $\boldsymbol{D(C)}$ adalah fungsi dekripsi, $\boldsymbol{d}$ adalah kunci privat untuk dekripsi *Ciphertext*, dan $\boldsymbol{C}$ adalah *Ciphertext*. Karena proses dekripsi juga menggunakan algoritma perpangkatan modulo (*Modular Exponentiation*), sehingga kompleksitasnya adalah $O((log\ n)^3)$ [9].

4.4.2. Analisis Algoritma McEliece

Sebelum melakukan enkripsi dan dekripsi, pembangkitan kunci dilakukan untuk menentukan kunci publik dan kunci privat. Untuk membangkitkan kunci pada sistem kriptografi McEliece, perlu dilakukan tahap-tahap sebagai berikut.

a. Alice memilih kode linear biner C(n,k) yang mampu memperbaiki error t. Kode ini harus mampu berperan sebagai algoritma decoding yang efisien, seperti goppa code dan membangkitkan matriks berukuran k x n matriks pembangkit G untuk kode C.
b. Alice memilih secara acak matriks biner S yang memiliki invers berukuran $k\ x\ k$.
c. Alice memilih secara acak matriks permutasi P berukuran $n\ x\ n$.
d. Alice menghitung matriks $\hat{G}$ dengan ukuran $k\ x\ n$, yaitu $\hat{G} = SGP$.
e. Kunci publik Alice adalah $(\hat{G}, t)$ dan kunci privatnya adalah (S, G, P).

Pada poin (a), kompleksitasnya adalah $O(1)$, karena berupa masukan. Poin (b) dan (c) hanya dilakukan pencarian invers matriks untuk mengecek matriks S dan P, sehingga kompleksitasnya adalah $O(n^3)$. Poin (d) diperlukan 2 perkalian matrix yang berukuran $k\ x\ k$, $k\ x\ n$, dan $n\ x\ n$, sehingga memiliki kompleksitas $O(k^2 n)$ untuk perkalian pertama dan $O(kn^2)$. Sehingga kompleksitas total diperoleh $O(n^3)$.

Setelah dilakukan pembangkitan kunci, proses enkripsi dan dekripsi dapat dilakukan. Enkripsi dilakukan dengan tahap-tahap sebagai berikut.

a. Bob melakukan encoding terhadap pesan m menjadi deretan biner dengan panjang k.
b. Bob menghitung vektor $c' = m\hat{G}$.
c. Bob membangkitkan vektor e sebesar n-bit secara acak yang mengandung t angka 1, agar hasil enkripsi semakin acak dan berdasarkan kode goppa yang bisa memperbaiki *error* hingga *t* [8]. ($t = weight$)
d. Bob menghitung pesan rahasia $c = c' + e$.

Saat enkipsi, pada poin (a), kompleksitasnya adalah $O(\log_2 m)$. Poin (b) hanya dilakukan perkalian matriks berukuran $1\ x\ k$ dan $k\ x\ n$, sehingga kompleksitasnya adalah $O(kn)$. Poin (d) diperlukan penjumlahan matrix yang masing-masing berukuran $1\ x\ n$, sehingga memiliki kompleksitas $O(n)$ Sehingga kompleksitas total diperoleh $O(kn^3)$.

Setelah pesan diterima, penerima melakukan proses dekripsi dengan tahap-tahap sebagai berikut.

a. Alice menghitung (P^{-1}), invers dari P.
b. Alice menghitung $\hat{c} = c(P^{-1})$.
c. Alice menggunakan algoritma decoding untuk kode C untuk proses decoding $\hat{c}$ menjadi $\hat{m}$, sehingga pesan kembali berukuran $1\ x\ k$.
d. Alice menghitung $m = \hat{m}S^{-1}$.

Saat dekripsi, pada poin (a), kompleksitasnya adalah $O(n^3)$ [11], karena pencarian invers matriks. Poin (b) hanya dilakukan perkalian matriks berukuran $1\ x\ n$ dan $n\ x\ n$, sehingga kompleksitasnya adalah $O(n^2)$. Poin (d) diperlukan penjumlahan matrix yang masing-masing berukuran $1\ x\ k$ dan $k\ x\ k$, sehingga memiliki kompleksitas $O(k^2)$ Sehingga kompleksitas total diperoleh $O(n^3)$.

4.5. Analisis Serangan Brute-Force pada Algoritma RSA

Tipe serangan ini berlaku untuk semua jenis algoritma kriptografi, baik kriptografi simetri maupun asimetri. Model serangan *brute-force* adalah dengan melakukan percobaan terhadap semua kemungkinan kunci pada sebuah sistem kriptografi. Oleh karena itu, performa serangan ini sangat bergantung kepada kemungkinan kunci dari sistem kriptografi yang ditargetkan. Algoritma RSA dikatakan aman jika penyerang kesulitan mencari faktor dari n, dan n merupakan bilangan yang besar.

Pada sistem RSA, *brute-force* bekerja dengan cara melakukan percobaan pada semua kemungkinan kunci privat dan penyerang telah mengetahui n dan e. Jika nilai p dan q sangat besar, maka akan dihasilkan nilai d yang cukup besar, karena d harus berada di rentang $1 < d < \phi(n)$, dan nilai $\phi(n)$berbanding lurus dengan nilai n, sehingga proses dekripsi *ciphertext* dengan cara $c^d \bmod n$ menghasilkan kompleksitas yang besar. [12].

4.6. Analisis Serangan Brute-Force pada Algoritma McEliece

Algoritma RSA dikatakan aman jika penyerang kesulitan mencari faktor dari kombinasi matriks biner pada $\widehat{\boldsymbol{G}}$, dan $\widehat{\boldsymbol{G}}$ merupakan matriks yang besar, sehingga *ciphertext* semakin panjang. Pada sistem McEliece, *brute-force* bekerja dengan cara melakukan percobaan pada semua kemungkinan kunci privat dan penyerang telah mengetahui $\widehat{\boldsymbol{G}}$ dan t. Jika nilai $\boldsymbol{n}$ dan $\boldsymbol{k}$ sangat besar, maka akan dihasilkan banyak kemungkinan kombinasi matriks $\boldsymbol{S}$, $\boldsymbol{G}$, dan $\boldsymbol{P}$, karena kombinasi matriks biner yang mungkin untuk semua matriks yaitu sebanyak $\mathbf{2^{kxk}}$ kombinasi matriks untuk $\boldsymbol{S}$, dan sebanyak $\mathbf{2^{kxn}}$ kombinasi matriks untuk $\boldsymbol{G}$, sehingga menghasilkan kompleksitas yang besar.

5. Kesimpulan

Berdasarkan hasil studi literatur, implementasi, dan pengujian sistem ini, maka kesimpulan yang didapat adalah RSA *Cryptosystem* memerlukan waktu yang lebih lama untuk melakukan enkripsi dan dekripsi daripada McEliece *Cryptosystem*. Waktu yang diperlukan pada kedua algoritma untuk proses enkripsi lebih lama daripada proses dekripsi. Gambar hasil enkripsi RSA memiliki histogram yang berbeda dengan gambar hasil enkripsi McEliece. Tingkat keacakan piksel pada gambar hasil enkripsi RSA lebih kecil daripada gambar hasil enkripsi McEliece dan gambar-gambar histogram kanan memiliki presentase perbedaan nilai entropi paling besar. McEliece memiliki kompleksitas lebih besar disbandingkan dengan RSA.

Saran untuk penelitian ke depan yaitu pencarian kompleksitas untuk operasi matriks biner, karena kompleksitas dari hasil analisis diperoleh dari studi literatur dan penulis tidak menemukan kompleksitas untuk operasi matriks biner.

Daftar Pustaka

[1]. El-Deen, A., El-Badawy, E. and Gobran, S., 2014. Digital image encryption based on RSA algorithm. J. Electron. Commun. Eng, 9(1), pp.69-73.

[2]. Singh, B.K. and Gupta, S.K., 2015. Grid-based Image Encryption using RSA. International Journal of Computer Applications, 115(1).

[3]. Chaladze, G. Kalatozishvili L. 2017. Linnaeus 5 Dataset for Machine Learning.

[4]. Dambra, A., Gaborit, P., Roussellet, M., Schrek, J. and Tafforeau, N., 2014. Improved Secure Implementation of McEliece Signature Schemes on Embedded Devices. IACR Cryptology ePrint Archive, 2014, p.163.

[5]. Rivest, R.L., Shamir, A. and Adleman, L., 1978. A method for obtaining digital signatures and public-key cryptosystems. Communications of the ACM, 21(2), pp.120-126.

[6]. Mceliece, R.J., 1978. A public-key cryptosystem based on algebraic. Coding Thv, 4244, pp.114-116

[7]. Zhao, G., Yang, X., Zhou, B. and Wei, W., 2010, July. RSA-based digital image encryption algorithm in wireless sensor networks. In 2010 2nd International Conference on Signal Processing Systems (Vol. 2, pp. V2-640). IEEE.

[8]. Stinson, D.R. and Paterson, M., 2018. Cryptography: Theory and Practice. CRC Press.

[9]. Taqa, A.Y. and Jalab, H.A., 2010. Increasing the reliability of skin detectors. Scientific Research and Essays, 5(17), pp.2480-2490.

[10]. Katz, J., Menezes, A.J., Van Oorschot, P.C. and Vanstone, S.A., 1996. Handbook of applied cryptography. CRC press.

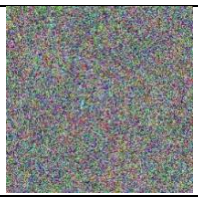
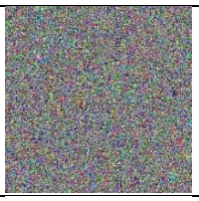
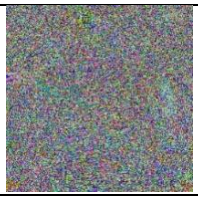
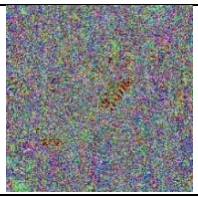
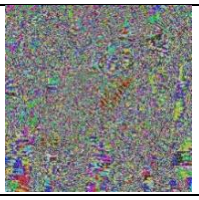
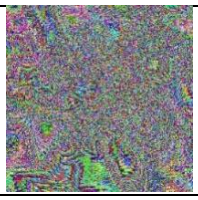
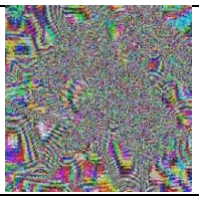
[11]. Gács, P. and Lovász, L., 1999. Complexity of algorithms. Lecture Notes, Yale University.

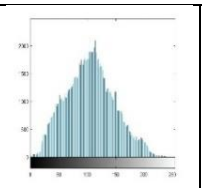
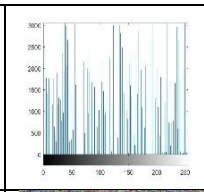
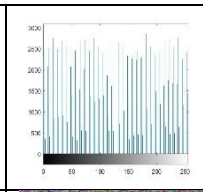
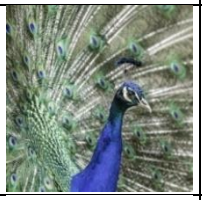
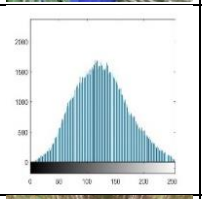
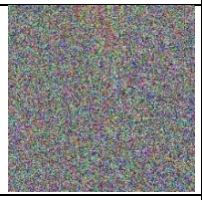
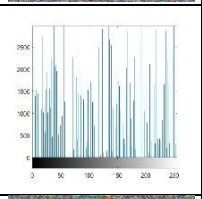
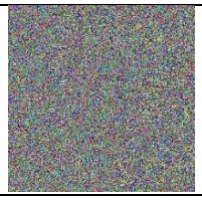
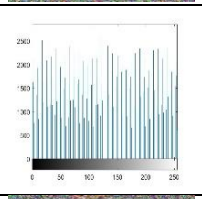
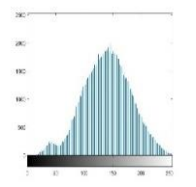
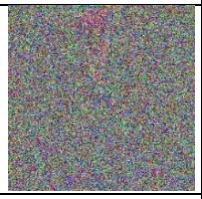
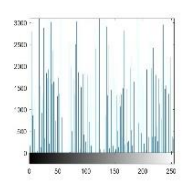
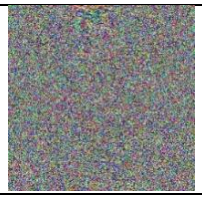
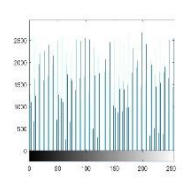
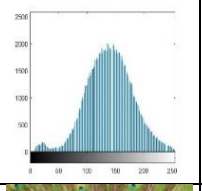
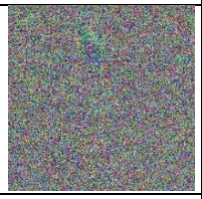
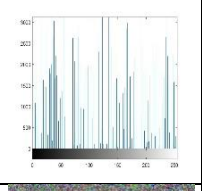
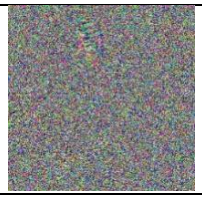
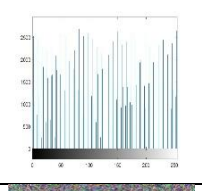
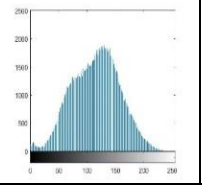
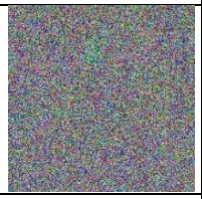
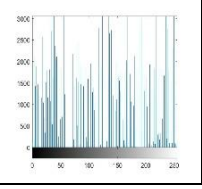
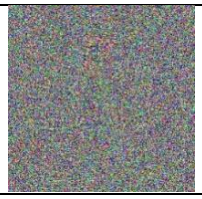
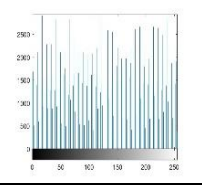
[12]. Pusparani, N.A., 2009. Analisis RSA dengan Penambahan Chinese Remainder Theorem untuk Mempercepat Proses Dekripsi. Skripsi. Institut Pertanian Bogor.

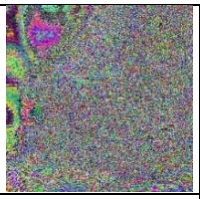
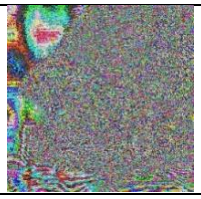
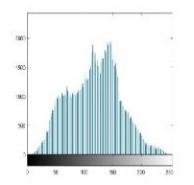
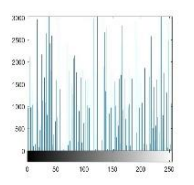
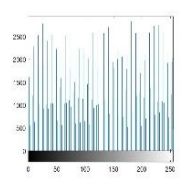
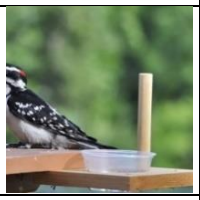
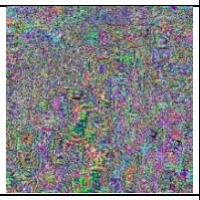
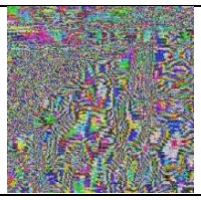
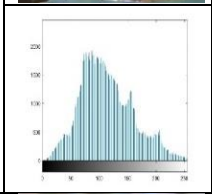
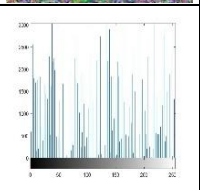
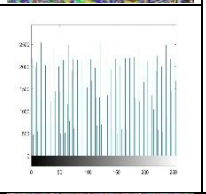
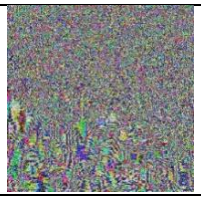
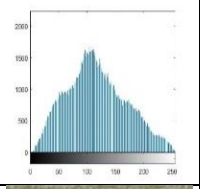
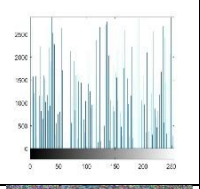
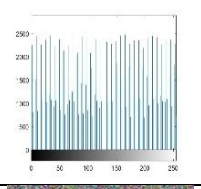
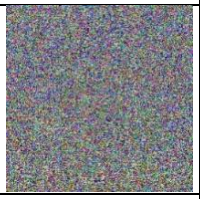
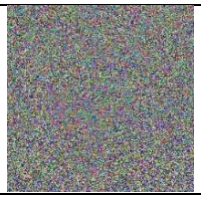
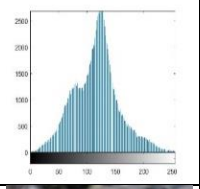
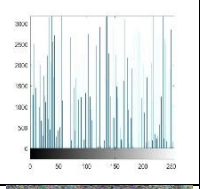
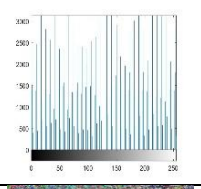
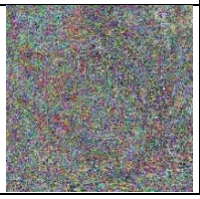
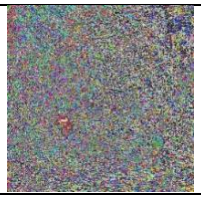
Lampiran

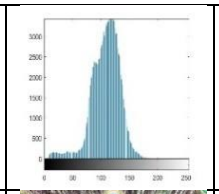
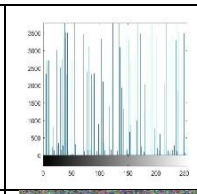
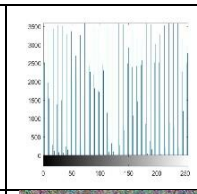
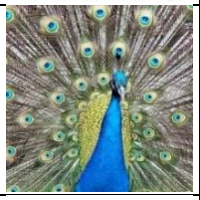
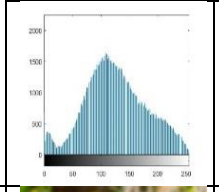
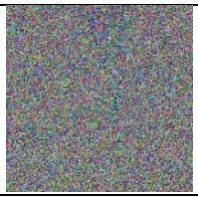
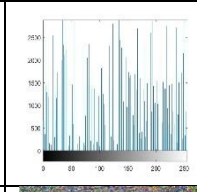
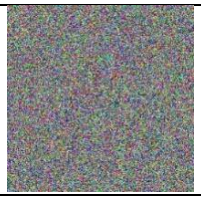
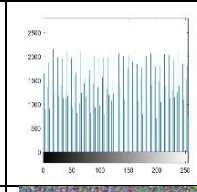
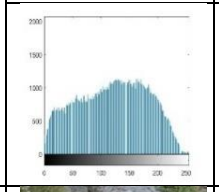
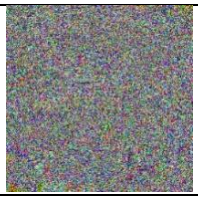
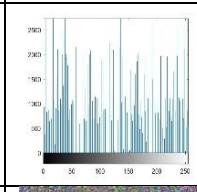
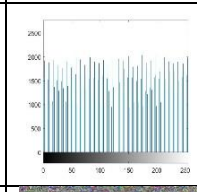
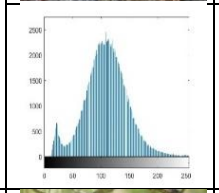
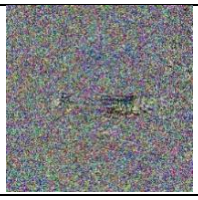
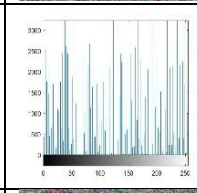
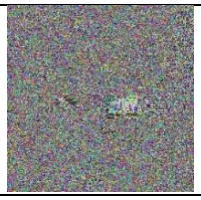
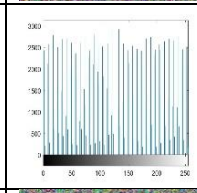
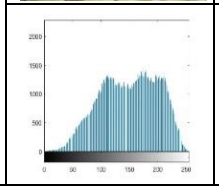
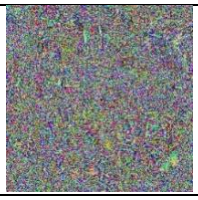
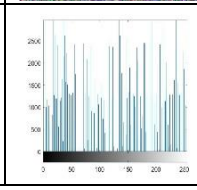
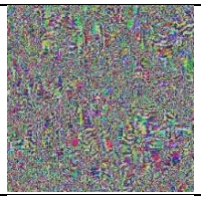
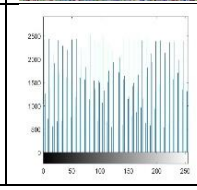
Pada bagian ini, dipaparkan 3 tabel hasil proses enkripsi dan perbandingan waktu komputasi enkripsi dan dekripsi antara RSA *Cryptosystem* dan McEliece *Cryptosystem*. Tabel 1 memaparkan hasil pengujian pada 20 gambar dengan histogram tengah. Tabel 2 memaparkan hasil pengujian pada 20 gambar dengan histogram kiri dan Tabel 3 memaparkan hasil pengujian pada 20 gambar dengan histogram kanan. Setelah itu, terdapat 3 tabel hasil pengujian yang dijadikan sebagai acuan dalam pembuatan grafik hasil pengujian gambar.

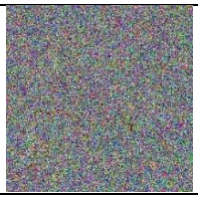
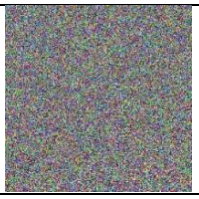
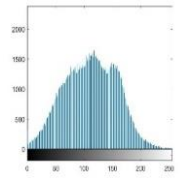
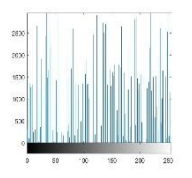
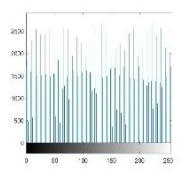
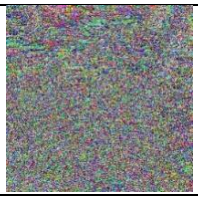
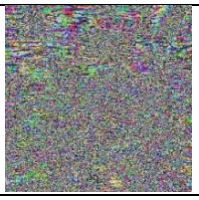
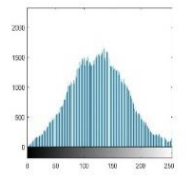
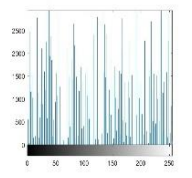
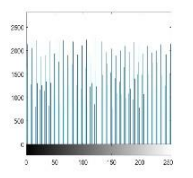
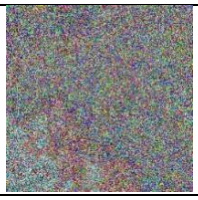
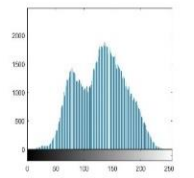
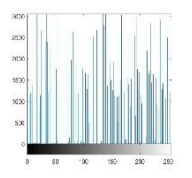
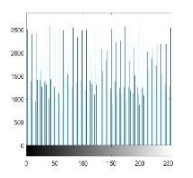
Tabel 1 Hasil Enkripsi dan Perbandingan Waktu Komputasi untuk Gambar Histogram Tengah

No	Gambar Asli	Hasil Enkripsi RSA	Hasil Enkripsi McEliece	Waktu Enkripsi (dalam s)		Waktu Dekripsi (dalam s)	
				RSA	McEliece	RSA	McEliece
1				497.788	304.433	254.895	76.895
2				473.511	345.828	258.209	83.472
3				463.526	341.793	270.378	96.086
4				446.366	325.117	263.55	83.408

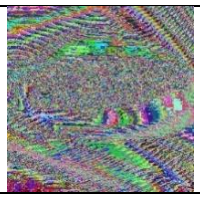
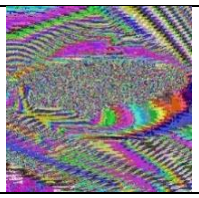
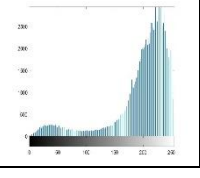
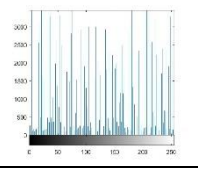
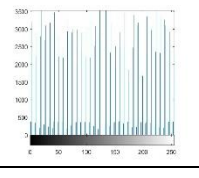
							
5	 	 	 	499.681	359.487	268.536	88.634
6	 	 	 	485.351	348.703	256.522	80.778
7	 	 	 	457.86	330.499	256.326	77.319
8	 	 	 	496.139	356.845	269.214	80.462

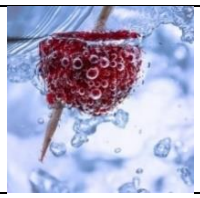
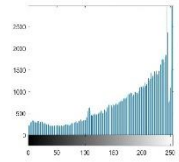
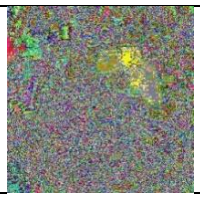
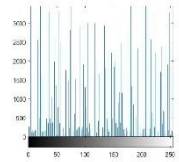
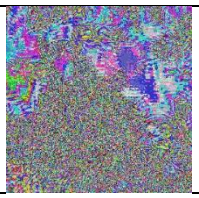
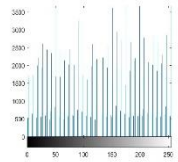
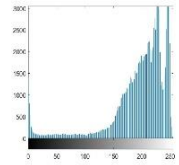
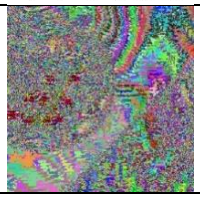
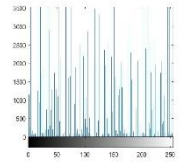
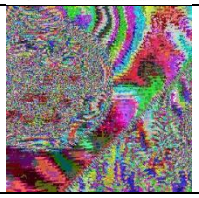
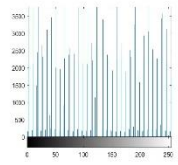
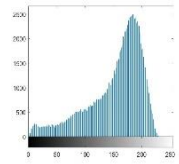
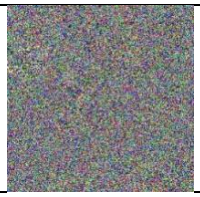
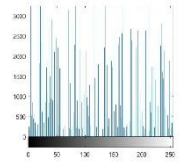
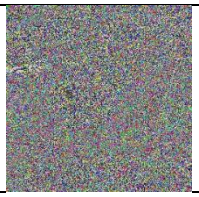
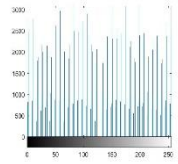
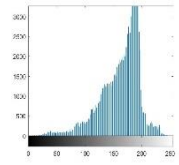
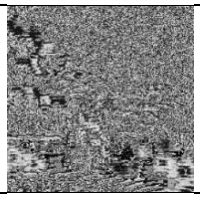
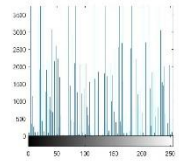
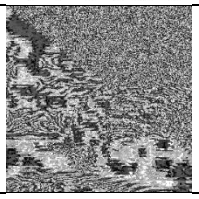
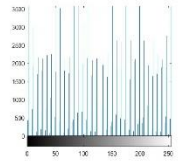
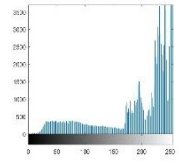
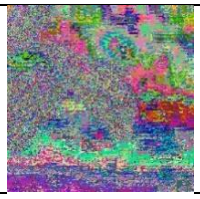
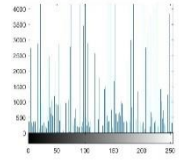
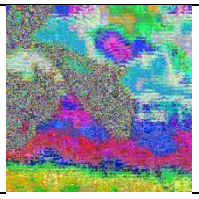
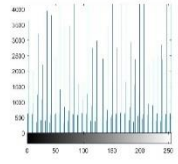
9				482.503	345.887	263.597	84.029
							
10				518.423	369.243	285.819	98.166
							
11				500.483	363.88	267.044	87.056
							
12				500.87	356.11	272.423	88.245
							
13				517.152	367.663	275.629	92.193

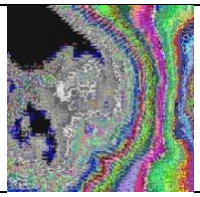
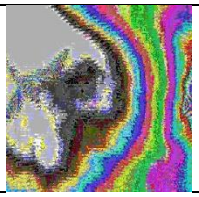
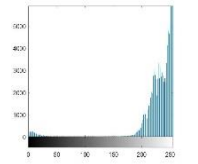
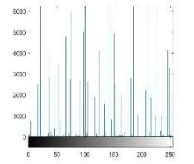
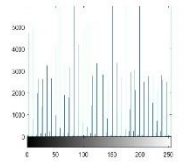
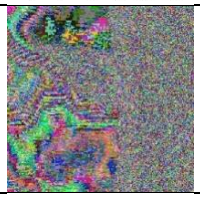
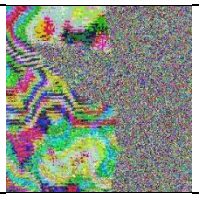
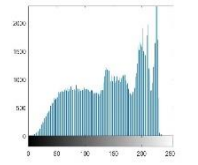
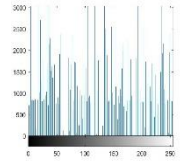
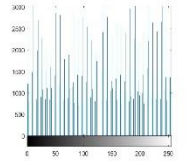
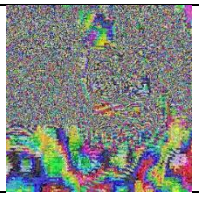
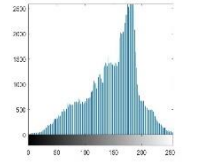
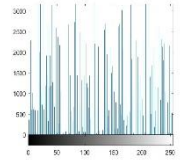
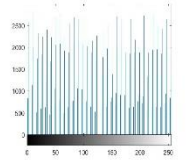
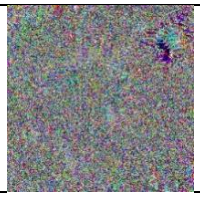
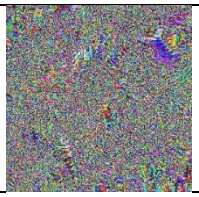
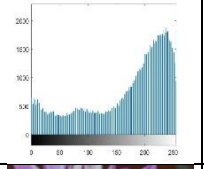
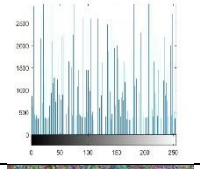
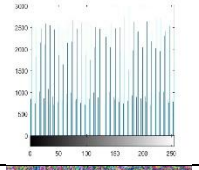
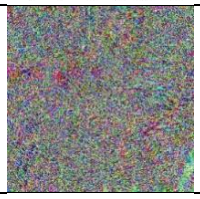
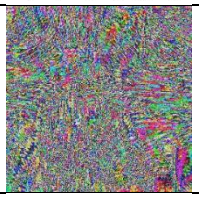
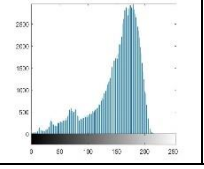
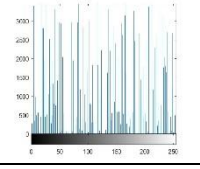
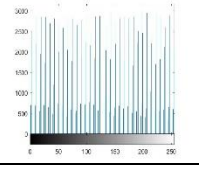
							
14	 	 	 	502.847	369.88	273.479	90.072
15	 	 	 	510.917	370.062	280.758	86.885
16	 	 	 	522.87	380.923	288.515	96.729
17	 	 	 	522.676	369.957	272.664	87.03

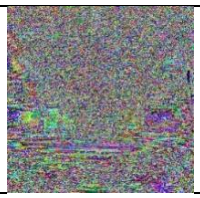
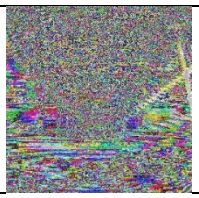
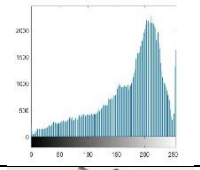
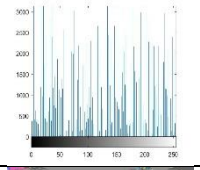
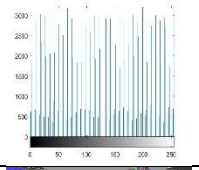
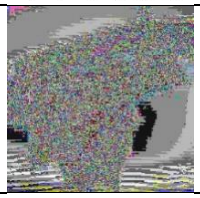
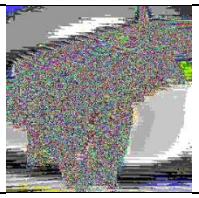
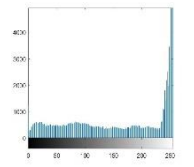
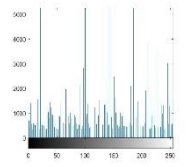
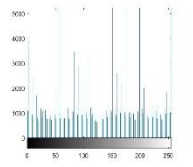
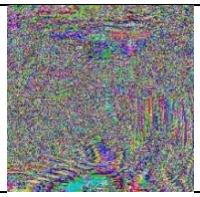
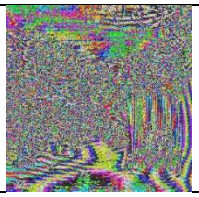
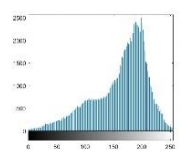
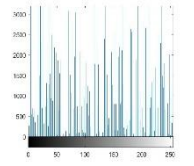
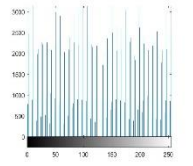
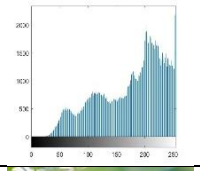
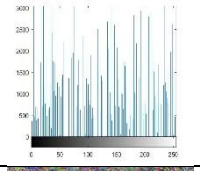
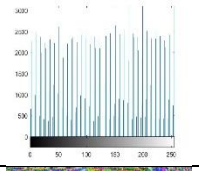
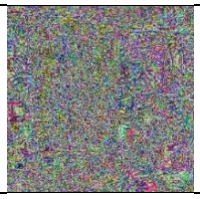
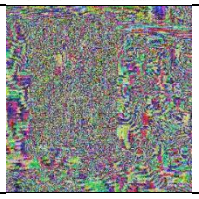
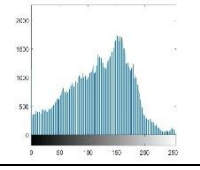
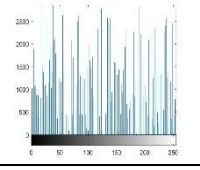
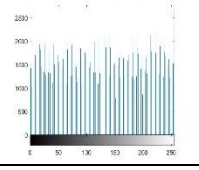
18				527.682	387.08	277.266	94.158
							
19				515.42	365.603	278.75	98.443
							
20				491.041	357.644	274.298	95.675
							
Rata-rata Waktu				496.6553	355.83185	270.3936	88.28675

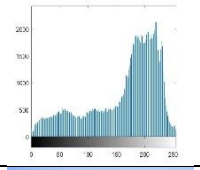
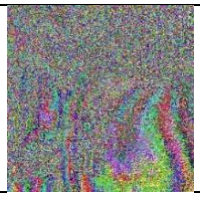
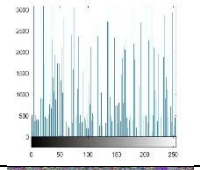
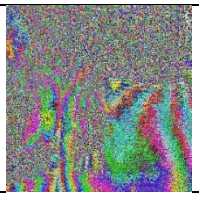
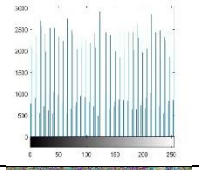
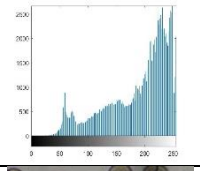
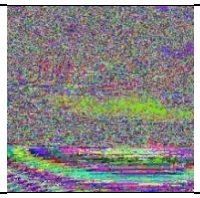
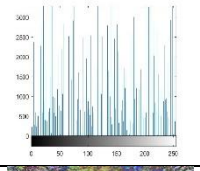
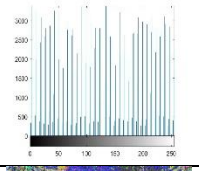
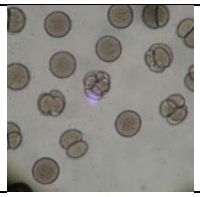
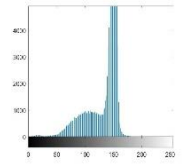
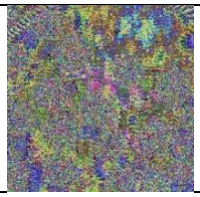
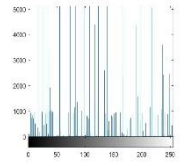
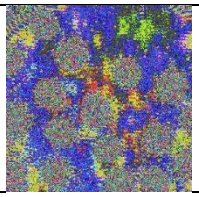
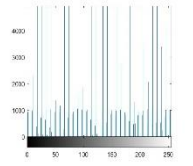
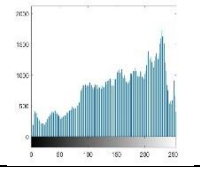
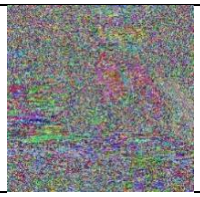
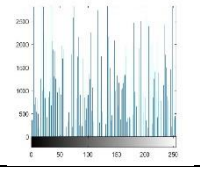
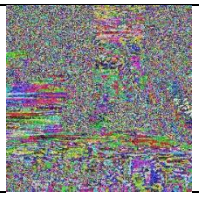
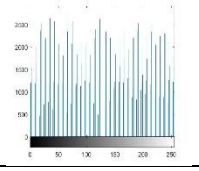
Tabel 2 Hasil Enkripsi dan Perbandingan Waktu Komputasi untuk Gambar Histogram Kiri

No	Gambar Asli	Hasil Enkripsi RSA	Hasil Enkripsi McEliece	Waktu Enkripsi (dalam s)		Waktu Dekripsi (dalam s)	
				RSA	McEliece	RSA	McEliece
1				517.528	365.072	270.077	91.366
							

2	 	 	 	471.441	343.766	268.888	84.452
3	 	 	 	512.847	367.95	271.679	89.403
4	 	 	 	506.41	364.296	268.296	87.946
5	 	 	 	504.95	357.987	264.649	82.926
6	 	 	 	504.677	369.985	273.223	84.013

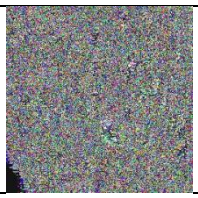
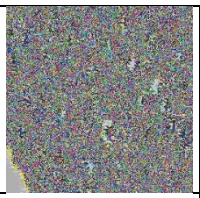
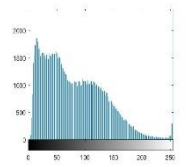
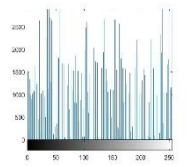
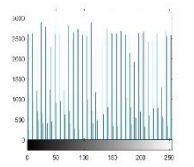
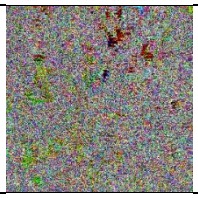
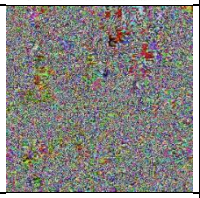
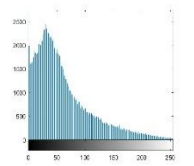
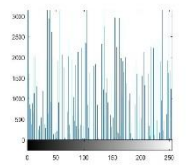
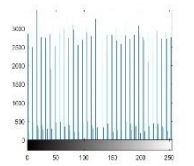
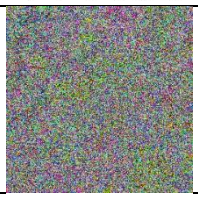
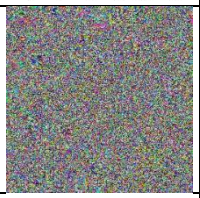
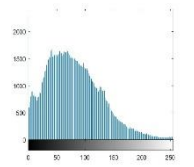
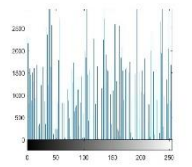
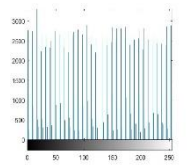
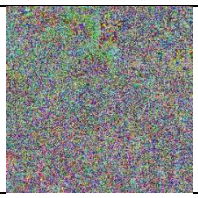
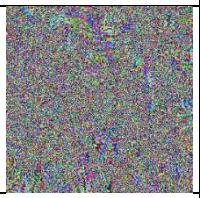
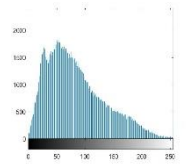
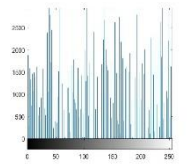
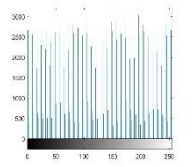
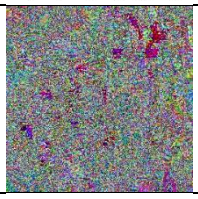
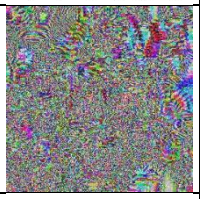
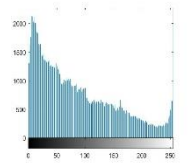
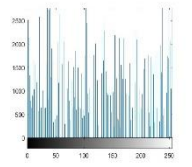
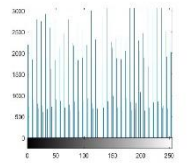
7				523.795	379.271	271.678	88.194
							
8				496.159	356.39	269.296	87.055
							
9				503.125	361.414	274.927	89.495
							
10				495.491	358.342	277.727	93.858
							
11				499.559	358.78	284.17	102.322
							

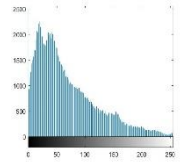
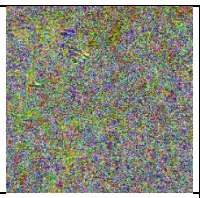
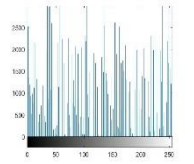
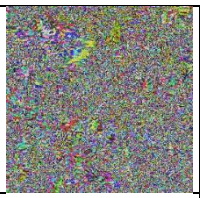
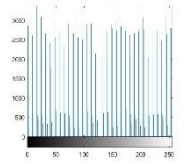
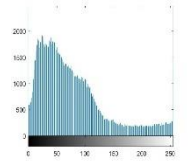
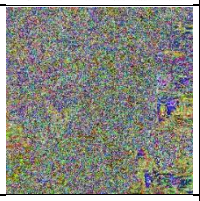
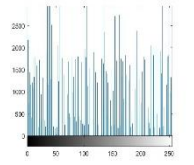
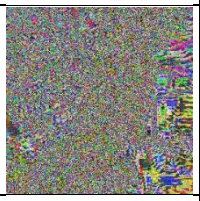
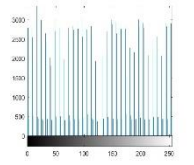
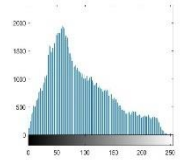
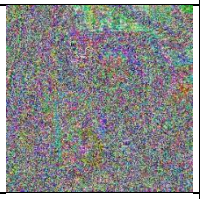
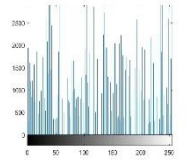
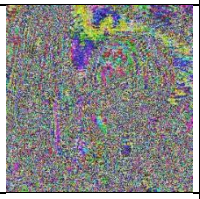
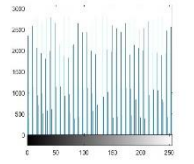
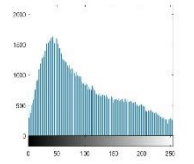
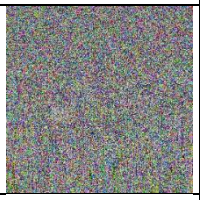
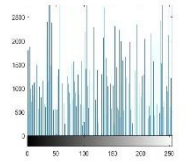
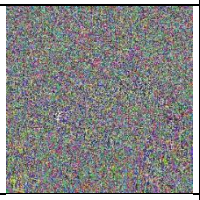
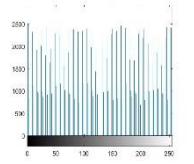
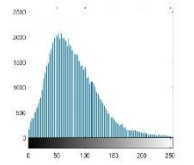
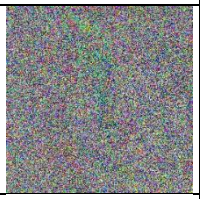
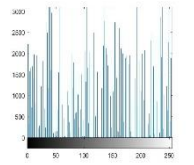
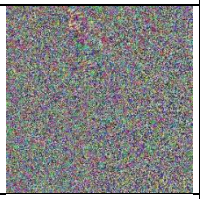
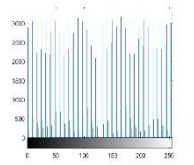
12							507.841	370.533	283.368	94.576
13							522.734	374.596	267.825	86.347
14							491.006	352.333	274.128	88.826
15							502.313	361.885	264.146	86.583
16							493.753	359.32	263.735	83.23

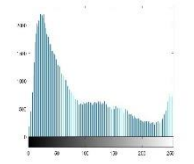
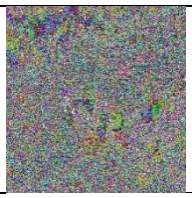
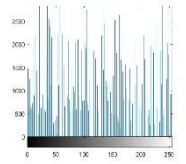
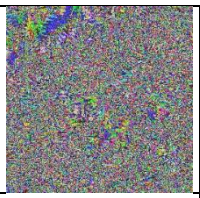
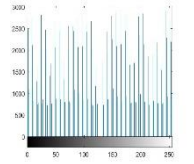
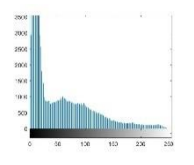
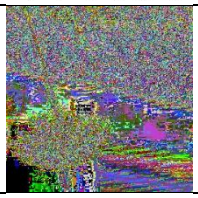
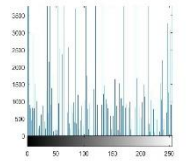
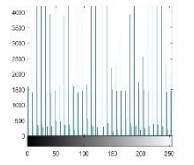
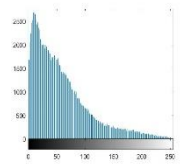
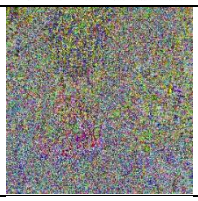
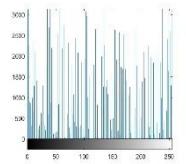
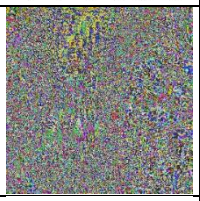
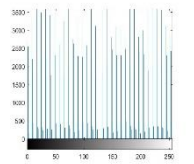
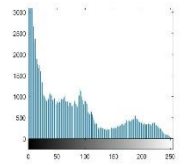
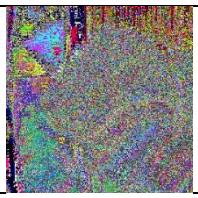
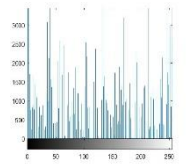
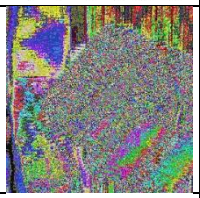
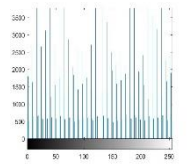
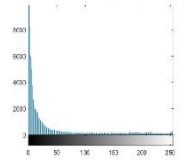
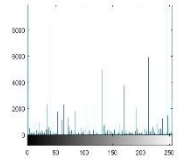
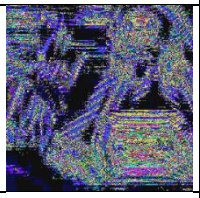
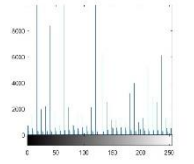
17	 	 	 	522.918	377.236	274.901	90.467
18	 	 	 	518.443	378.816	275.365	90.54
19	 	 	 	537.063	389.477	265.41	84.062
20	 	 	 	511.882	371.085	273.763	89.346
Rata-rata waktu				507.19675	365.9267	271.86255	88.75035

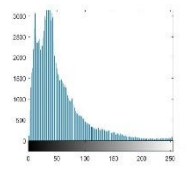
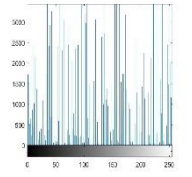
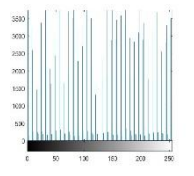
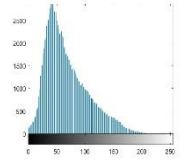
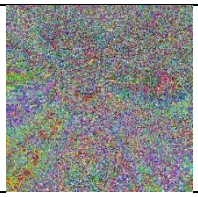
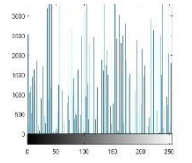
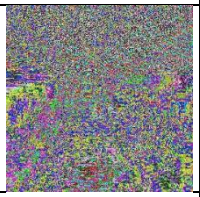
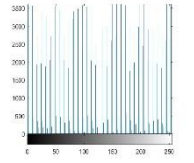
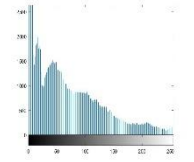
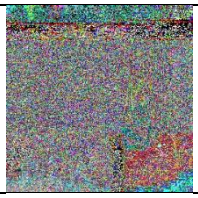
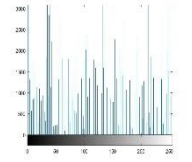
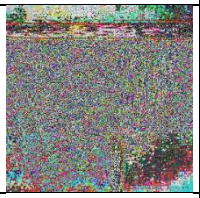
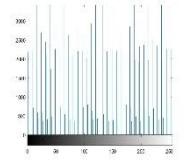
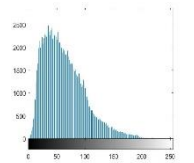
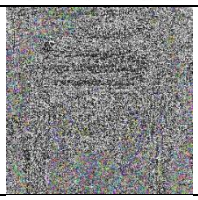
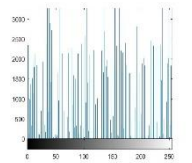
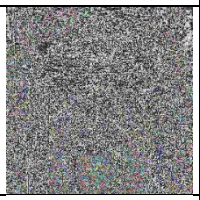
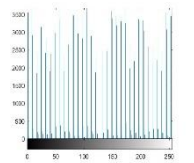
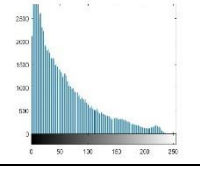
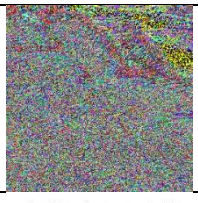
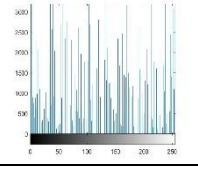
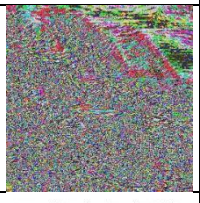
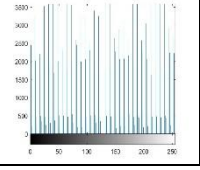
Tabel 3 Hasil Enkripsi dan Perbandingan Waktu Komputasi untuk Gambar Histogram Kanan

No	Gambar Asli	Hasil Enkripsi RSA	Hasil Enkripsi McEliece	Waktu Enkripsi (dalam s)		Waktu Dekripsi (dalam s)	
				RSA	McEliece	RSA	McEliece

1							528.495	381.865	270.876	88.541
2							544.629	396.241	273.356	89.191
3							497.178	358.378	271.122	88.983
4							538.686	385.926	262.932	81.821
5							507.066	363.186	290.924	90.094

6	 	 	 	517.254	370.37	271.343	91.133
7	 	 	 	504.404	369.91	261.886	81.639
8	 	 	 	542.137	333.188	295.727	72.285
9	 	 	 	514.002	362.556	272.754	92.702
10	 	 	 	519.78	370.511	274.055	93.122

11	 	 	 	517.701	372.607	281.866	94.37
12	 	 	 	495.517	357.978	275.089	91.405
13	 	 	 	490.424	357.773	263.623	84.088
14	 	 	 	494.674	356.09	267.947	87.543
15	 	 	 	511.665	375.086	268.95	86.849

16	 	 	 	509.271	370.554	262.594	82.63
17	 	 	 	499.612	354.939	272.61	97.229
18	 	 	 	509.434	366.145	268.837	89.851
19	 	 	 	503.819	357.688	263.155	86.114
20	 	 	 	474.62	344.335	261.668	78.717

Rata-rata Waktu	511.0184	365.2663	271.5657	87.41535

Hasil pengujian untuk proses enkripsi pada gambar dapat dilihat pada tabel 1 untuk gambar-gambar dengan histogram tengah, tabel 2 untuk gambar-gambar dengan histogram kiri, dan tabel 3 untuk gambar-gambar dengan histogram kanan.

Tabel 4 Waktu Komputasi Enkripsi dan Dekripsi untuk Gambar Histogram Tengah

No	Gambar Asli	Waktu Enkripsi		Waktu Dekripsi	
		RSA	McEliece	RSA	McEliece
1	center 1	497.788	304.433	254.895	76.895
2	center 2	473.511	345.828	258.209	83.472
3	center 3	463.526	341.793	270.378	96.086
4	center 4	446.366	325.117	263.55	83.408
5	center 5	499.681	359.487	268.536	88.634
6	center 6	485.351	348.703	256.522	80.778
7	center 7	457.86	330.499	256.326	77.319
8	center 8	496.139	356.845	269.214	80.462
9	center 9	482.503	345.887	263.597	84.029
10	center 10	518.423	369.243	285.819	98.166
11	center 11	500.483	363.88	267.044	87.056
12	center 12	500.87	356.11	272.423	88.245
13	center 13	517.152	367.663	275.629	92.193
14	center 14	502.847	369.88	273.479	90.072
15	center 15	510.917	370.062	280.758	86.885
16	center 16	522.87	380.923	288.515	96.729
17	center 17	522.676	369.957	272.664	87.03
18	center 18	527.682	387.08	277.266	94.158
19	center 19	515.42	365.603	278.75	98.443
20	center 20	491.041	357.644	274.298	95.675
Rata-rata Waktu		496.6553	355.83185	270.3936	88.28675

Tabel 5 Waktu Komputasi Enkripsi dan Dekripsi untuk Gambar Histogram Kiri

No	Gambar Asli	Waktu Enkripsi		Waktu Dekripsi	
		RSA	McEliece	RSA	McEliece
1	left 1	517.528	365.072	270.077	91.366
2	left 2	471.441	343.766	268.888	84.452
3	left 3	512.847	367.95	271.679	89.403
4	left 4	506.41	364.296	268.296	87.946
5	left 5	504.95	357.987	264.649	82.926
6	left 6	504.677	369.985	273.223	84.013
7	left 7	523.795	379.271	271.678	88.194
8	left 8	496.159	356.39	269.296	87.055
9	left 9	503.125	361.414	274.927	89.495
10	left 10	495.491	358.342	277.727	93.858
11	left 11	499.559	358.78	284.17	102.322

12	left 12	507.841	370.533	283.368	94.576
13	left 13	522.734	374.596	267.825	86.347
14	left 14	491.006	352.333	274.128	88.826
15	left 15	502.313	361.885	264.146	86.583
16	left 16	493.753	359.32	263.735	83.23
17	left 17	522.918	377.236	274.901	90.467
18	left 18	518.443	378.816	275.365	90.54
19	left 19	537.063	389.477	265.41	84.062
20	left 20	511.882	371.085	273.763	89.346
Rata-rata Waktu		507.19675	365.9267	271.86255	88.75035

Tabel 6 Waktu Komputasi Enkripsi dan Dekripsi untuk Gambar Histogram Kanan

No	Gambar Asli	Waktu Enkripsi		Waktu Dekripsi	
		RSA	McEliece	RSA	McEliece
1	right 1	528.495	381.865	270.876	88.541
2	right 2	544.629	396.241	273.356	89.191
3	right 3	497.178	358.378	271.122	88.983
4	right 4	538.686	385.926	262.932	81.821
5	right 5	507.066	363.186	290.924	90.094
6	right 6	517.254	370.37	271.343	91.133
7	right 7	504.404	369.91	261.886	81.639
8	right 8	542.137	333.188	295.727	72.285
9	right 9	514.002	362.556	272.754	92.702
10	right 10	519.78	370.511	274.055	93.122
11	right 11	517.701	372.607	281.866	94.37
12	right 12	495.517	357.978	275.089	91.405
13	right 13	490.424	357.773	263.623	84.088
14	right 14	494.674	356.09	267.947	87.543
15	right 15	511.665	375.086	268.95	86.849
16	right 16	509.271	370.554	262.594	82.63
17	right 17	499.612	354.939	272.61	97.229
18	right 18	509.434	366.145	268.837	89.851
19	right 19	503.819	357.688	263.155	86.114
20	right 20	474.62	344.335	261.668	78.717
Rata-rata Waktu		511.0184	365.2663	271.5657	87.41535