Algoritma *Elgamal* dengan Pertukaran Kunci *Diffie Hellman* pada Aplikasi Keamanan Citra Sidik Jari Berbasis Android

Nurul Yalisa[1], Muhammad Arhami[2], Azhar[3]

[1,2,3] *Jurusan Teknologi Informasi dan Komputer Politeknik Negeri Lhokseumawe*
Jln. B.Aceh Medan Km.280 Buketrata 24301 INDONESIA

nurulyalisa9@gmail.com[1],
muhammad.arhami@gmail.com[2],
tgk.azhar@yahoo.com[3]

Abstrak- Kemajuan teknologi informasi dan telekomunikasi sudah sangat pesat. Berbagai macam data sudah dapat dikirimkan secara global namun dalam implementasinya pengiriman data secara global tidak selalu aman. Berbagai macam cara dikembangkan untuk pengamanan data, salah satunya dengan menggunakan kriptografi. Kriptografi yaitu cara mengamankan data dengan menggunakan transformasi data sehingga data yang dihasilkan tidak dapat di mengerti oleh pihak ketiga. Tugas akhir ini membuat aplikasi pengamanan data pada citra sidik jari berbasis Android dengan menggunakan algoritma *Elgamal* serta *Diffie Hellman* sebagai metode pertukaran kunci. Algortima *Diffie Hellman* digunakan untuk proses pertukaran dan pembangkitan kunci sedangkan algoritma *elgamal* digunakan untuk proses enkripsi dan dekripsi. Citra yang akan dienkripsi pada aplikasi ini adalah citra sidik jari berukuran 180 x 230 piksel. Proses enkripsi akan menghasilkan *chiper* citra berukuran 460 x 360 piksel atau berukuran 2 kali dari ukuran citra aslinya, selanjutnya di dekripsi untuk menghasilkan *plain* citra (citra asli). Dari 30 sampel yang di uji dengan menggunakan kunci 16 bit, hasil enkripsi dan dekripsi berhasil dilakukan dengan waktu rata-rata waktu enkripsi 102,482 detik dan rata-rata waktu dekripsi 34,151 detik.

Kata kunci: *Elgamal*, *Diffie Hellman, Chiper,* enkripsi, dekripsi

Abstract- *Advances in information technology and telecommunications have been very rapid. Various data can already be sent globally but in the implementation of data delivery globally is not always safe. Various ways developed for data security, one of them by using cryptography. Cryptography is a way of securing data by using data transformation so that the data generated can not be understood by third parties. This final project creates data security applications on Android based fingerprint image by using Elgamal algorithm and Diffie Hellman as the key exchange method. Algortima Diffie Hellman is used for the exchange and generation of keys while the elgamal algorithm is used for encryption and decryption processes. The image to be encrypted in this app is a 180 x 230 pixel fingerprint image. The encryption process will produce an image chip measuring 460 x 360 pixels or 2 times the size of the original image, then decrypted to produce plain image (original image). From 30 samples tested using 16 bit key, the result of encryption and decryption was done with the average time of encryption time 102,482 second and mean time of decryption 34,151 second.*

Keyword: Elgamal, Diffie Hellman, Chiper, encrypted, decrypted

I. PENDAHULUAN

Pada Era Globalisasi kini perkembangan teknologi informasi berkembang pesat. Segala macam data kini sudah dapat dikirimkan dengan cepat, murah dan jangkauan wilayah yang tidak terbatas. Pada implementasinya pengiriman data tidak selalu aman, banyak terjadinya penyadapan oleh pihak-pihak yang tak bertanggung jawab sehingga mengakibatkan pemalsuan data, manipulasi data, kerusakan data dan lain-lain, karenanya dibutuhkan pengamanan data agar setiap data yang dikirimkan tetap terjaga keaslian dan keautentikasiannya.

Banyak data – data yang perlu pengamanan, salah satunya pengamanan data citra sidik jari. Sidik jari dalam dunia modern menunjukkan identitas dari seseorang, karena sidik jari merupakan suatu hal yang unik dimana tiap orang tidak pernah memiliki sidik jari yang sama satu sama lain. Namun sidik jari sering disalahgunakan terlihat dari banyaknya kasus pemalsuan sidik jari yang terjadi pada saat ini, misalnya penggunaan sidik jari untuk masuk ke hak akses orang lain pada suatu instansi tertentu.

Berbagai macam cara dikembangkan untuk pengamanan data, salah satunya dengan menggunakan kriptografi. Kriptografi yaitu cara yang dilakukan untuk mengamankan data dengan menggunakan transformasi data sehingga data yang dihasilkan tidak dapat di mengerti oleh pihak ketiga. Transformasi ini dapat memberikan solusi pada dua masalah keamanan data, yaitu masalah privasi dan keautentikan data.

Kriptografi adalah ilmu dan seni untuk menjaga kerahasiaan suatu informasi. Dalam kriptografi terdapat dua konsep utama yaitu proses enkripsi dan dekripsi. Enkripsi adalah proses dimana informasi atau data yang hendak dikirim diubah menjadi bentuk yang tidak dapat dikenali oleh orang awam. Dekripsi adalah kebalikan dari enkripsi yaitu mengubah bentuk yang tidak dikenali menjadi informasi awal) [1].

Pada penelitian [2] melakukan penelitian tugas akhir dengan judul "Rancang bangun aplikasi kriptografi untuk pengamanan citra RGB 24 bit dengan menggunakan metode *elgamal*". Hasil dari sistem ini adalah citra RGB 24 bit tidak dikenali lagi (sudah teracak) dari citra aslinya. Dengan adanya sistem ini dapat mengamankan data gambar dari pihak-pihak yang tidak bertanggung jawab.

Pada penelitian lain [3] telah melakukan penelitian yang telah dijurnalkan dengan judul "Aplikasi Kriptografi Asimetris dengan Metode *Diffie Hellman* dan Algoritma *Elgamal* untuk Kemanan Teks". Hasil penelitian aplikasi yang dibangun dapat menerapkan metode pertukaran kunci *Diffie Hellman* dan menghasilkan kunci baru dan aplikasi ini dapat melakukan enkripsi deskripsi *elgamal* dengan menggunakan kunci yang telah dibangkitkan dengan menggunakan metode *Diffie Hellman.*

Penelitian kali ini dirancang aplikasi keamanan citra sidik jari berbasis android dengan menggunakan algoritma *Elgamal* serta *Diffie Hellman* sebagai metode pertukaran kunci. Penggabungan antara dua metode yaitu *Diffie Hellman* dan algoritma *Elgamal* yang mana penggunaan *Diffie Hellman* pada kasus ini untuk pertukaran kunci, dimana pengirim dan penerima akan memiliki angka-angka rahasia tertentu kemudian saling bertukar untuk dihitung kunci publik dan kunci privat. Setelah memperoleh kunci publik dan kunci privat akan dilanjutkan dengan proses perhitungan untuk enkripsi dan dekripsi dengan menggunakan algoritma *elgamal*.

II. METODOLOGI PENELITIAN

A. Analisis Permasalahan

Analisis permasalahan dalam penelitian ini adalah penggabungan antara metode *Diffie Hellman* untuk proses pembangikitan kunci rahasia, pertukaran kunci, serta pembangkitan kunci public serta kunci privat dan algortima *Elgamal* untuk proses enkripsi dan dekripsi file citra sidik jari nantinya.

B. Pembangkitan Kunci dengan Menggunakan Metode Diffie Hellman

Algoritma kunci publik diterbitkan pertama kali dalam makalah *Diffie* dan *Hellman* yang didefinisikan sebagai kriptografi kunci publik dan biasanya disebut sebagai *Diffie-Hellman Key Exchange* (pertukaran kunci) atau Protokol *Diffie-Hellman.* Tujuan dari algoritma ini adalah untuk memungkinkan dua pengguna saling bertukar kunci secara aman, kemudian dapat digunakan untuk enkripsi dan dekripsi pesan berikutnya [4].

Algoritma ini tidak berdasarkan pada proses enkripsi dan dekripsi, melainkan lebih kepada proses matematika yang dilakukan untuk menghasilkan kunci rahasia yang dapat disebarkan secara bebas tanpa harus khawatir karena kunci rahasia tersebut hanya dapat didekripsi hanya oleh pengirim dan penerima pesan. [5].

Besaran besaran yang digunakan dalam algoritma *Diffie Hellman* adalah sebagai berikut :

1. Bilangan prima p bersifat tidak rahasia
2. Bilangan bulat acak g (g < p) bersifat tidak rahasia
3. Bilangan bulat acak x ($1 \leq x \leq p-2$) bersifat rahasia.
4. Bilangan x pengirim dan y penerima bersifat rahasia.
5. Bilangan A dan B yang bersifat tidak rahasia
6. Bilangan Y_a dan Y_b yang bersifat rahasia

1. Tingkat Keamanan Algoritma Diffie Hellman

Tingkat keamanan dari algoritma ini tinggi, jika nilai p dan g dipilih secara benar. Karena untuk mengetahui atau menebak nilai rahasia yang dimiliki oleh penerima dan pengirim harus menyelesaikan persamaan Diffie-Hellman terlebih dahulu. Ini merupakan masalah logaritma diskrit yang perhitungan tersebut tidak dapat diselesaikan untuk nilai bilangan p yang sangat besar. Menghitung logaritma diskrit dari bilangan modulo p memakan waktu yang kurang lebih sama seperti dengan memfaktorkan bilangan non prima menjadi faktor primanya [5].

2. Proses Pertukaran kunci

Pada proses pertukaran kunci menentukan nilai bilangan prima yang besar, *p* dan bilangan integer yang tidak melebihi dari nilai *g,* Kedua bilangan tersebut dapat diketahui secara publik. Selanjutnya pilih sebuah bilangan acak oleh pengirim yakni ***x***, bilangan ini tidak boleh diketahui oleh orang lain dan juga sebuah bilangan acak oleh penerima yakni ***y***, bilangan ini tidak boleh diketahui oleh orang lain. Berikut persamaan untuk menghitung kunci public A dan B:

$$A = g^x \bmod p. \quad (1)$$

$$B = g^y \bmod p. \quad (2)$$

Keterangan :

A = Kunci publik A
B = Kunci Publik B
g = Kunci public g
p = Kunci public p
x = kunci privat pengirim
y = kunci privat penerima

Lakukan pertukaran bilangan A dan B terhadap pengirim dan penerima.

$$Y_a = B^x \bmod p. \quad (3)$$

$$Y_b = A^y \bmod p. \quad (4)$$

Keterangan :

Y_a = Kunci privat Y_a
Y_b = Kunci privat Y_b
Y = Kunci privat Y

Berdasarkan hukum aljabar nilai Y_a sama dengan Y_b atau bisa disebut $Y_a=Y_b=Y$. Sehingga pengirim dan penerima tersebut mengetahui kunci rahasia tersebut "Y". Untuk memudahkan, Y untuk kunci privat sering disebut dengan Y_{privat}, lalu hitung kunci publik untuk proses enkripsi dengan persamaan :

$$Y_{publik} = g^{privat} \bmod p. \quad (5)$$

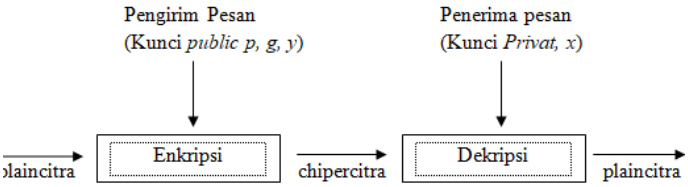
Keterangan :
Y_{publik} = kunci Publik Y

Langkah-langkah dalam pertukaran kunci Diffie-Hellman adalah sebagai berikut:

1. Pilih bilangan prima yang besar, *p = 331* dan bilangan integer yang tidak melebihi dari nilai *p*, *g = 103* biasa disebut bilangan basis atau generator. Kedua bilangan tersebut dapat diketahui secara publik.
2. Pilih sebuah bilangan acak oleh pengirim, *x = 41*, bilangan ini tidak boleh diketahui oleh orang lain.
3. Pilih sebuah bilangan acak oleh penerima, *y = 53*, bilangan ini tidak boleh diketahui oleh orang lain.
4. Pengirim menghitung kunci publik $A = g^x \bmod p = 103^{41} \bmod 331 = 25$. Bilangan A ini dapat diketahui secara publik.
5. Penerima menghitung kunci public $B = g^y \bmod p = 103^{53} \bmod 331 = 76$. Bilangan B ini dapat diketahui secara publik.
6. Lakukan pertukaran bilangan A dan B terhadap pengirim dan penerima. Pengirim mendapat kunci publik B = 76 dan penerima mendapat kunci publik A = 25
7. Lalu Pengirim menghitung kunci privat $Y_a = B^x \bmod p = 76^{41} \bmod 331 = 26$.
8. Penerima menghitung kunci privat $Y_b = A^y \bmod p = 25^{53} \bmod 331 = .26$
9. Berdasarkan hukum aljabar nilai Y_a sama dengan Y_b atau bisa disebut $Y_a = Y_b = Y = 26$. Sehingga pengirim dan penerima tersebut mengetahui kunci rahasia tersebut "Y". kunci privat = 26
10. lalu hitung kunci public untuk proses enkripsi dengan persamaan
$Y_{publik} = g^{privat} \bmod p = 103^{26} \bmod 331 = 21$.

C. *Proses Enkripsi dan Dekripsi dengan Menggunakan Algoritma Elgamal*

Algoritma elgamal mempunyai kunci *public* berupa tiga pasang bilangan (p,g,y) dan kunci rahasia berupa satu bilangan (x). Algoritma ini mempunyai kerugian pada chipertextnya yang mempunyai panjang dua kali lipat dari plainteksnya. Akan tetapi, algoritma ini mempunyai kelebihan pada enkripsi. Untuk plainteks yang sama, algoritma ini memberikan chiperteks yang berbeda (dengan kepastian yang dekat) setiap kali plainteks di enkripsi [6].



Gambar 1 Blok Diagram Elgamal

Besaran besaran yang digunakan dalam algoritma kriptografi elgamal adalah sebagai berikut :

1. Bilangan prima p bersifat tidak rahasia
2. Bilangan bulat acak g (g < p) bersifat tidak rahasia
3. Bilangan bulat acak x ($1 \leq x \leq p-2$) bersifat rahasia.
4. Bilangan y bersifat tidak rahasia.
5. m (plaincitra) bersifat rahasia merupakan pesan asli yang digunakan untuk data sumber dalam proses enkripsi dan merupakan data hasil pada proses dekripsi.
6. a dan b (ciphercitra) bersifat tidak rahasia

Proses enkripsi

Pada proses enkripsi dilakukan dengan menyusun nilai-nilai intensitas sesuai blok-blok pada *pixel* citra. Nilai-nilai ini yang disebut nilai m (*plain citra*). Nilai m harus masih berada didalam range 0 sampai p-1.

$$a = g^k \bmod p \quad (6)$$
$$b = y^k m \bmod p \quad (7)$$

keterangan:
g = kunci publik g (merupakan bilangan prima)
k = bilangan acak k dengan syarat $0 \geq k \leq p-2$
p = kunci publik p (merupakan bilangan prima)
a = chipercitra urutan ganjil
b = chipercitra urutan genap
y = kunci publik y

User A (pengirim pesan) sebelumnya membangkitkan kunci privat dan kunci public, begitu juga dengan user B (penerima pesan). Kunci publik digunakan untuk proses enkripsi dan kunci privat digunakan untuk proses dekripsi. Untuk setiap piksel citra yang dienkripsi akan menggunakan bilangan k yang berbeda-beda. Satu piksel citra yang direpresentasikan dengan menggunakan bilangan bulat yang akan di susun menjadi blok-blok m1, m2,…., sedemikian sehingga setiap blok merepresentasikan nilai di dalam selang [0, p – 1]. Hasil dari proses enkripsi ini akan menghasilkan kode dalam bentuk blok yang terdiri atas dua nilai (a,b).

Langkah-langkah yang dilakukan oleh user A (pengirim) adalah:

1. Mengubah citra sidik jari tersebut ke dalam bentuk matrik piksel citra. Sesuai dengan nilai RGB dari setiap piksel citra. Dalam tabel matrik piksel 1 x 1 "P" = 331, nilai piksel tersebut sebagai nilai *m = 192,192,192*
2. Menentukan nilai acak *k* yang dalam hal ini $0 \leq k \leq p - 2$, misalkan nilai *k* adalah 59

3. Melakukan enkripsi terhadap matrik piksel "P", enkripsi dilakukan dengan menghitung nilai *a* menggunakan persamaan (2.7) sehingga diperoleh $a = 103^{59} \pmod{331}$ adalah 20, serta menghitung nilai *b* menggunakan persamaan (2.8) sehingga di peroleh $b = 21^{59} \pmod{192}$ adalah 220. Susunan ciphercitra yang dihasilkan adalah 20,220.
4. Pesan yang dikirim kepada user B dalam bentuk ciphercitra (20,20,20, 220,220,220). Di konversi kembali menjadi piksel dengan nilai RGB untuk matrik 20,20,20 dan 220,220,220

Proses dekripsi

Pada proses deskripsi digunakan kunci privat dan public *p* untuk mendeskripsi *a* dan *b* menjadi *plain* citra m dengan persamaan.

$$(a^{privat})^{-1} = a^{p-1-privat} \bmod p \quad (8)$$

$$M = b(a^{privat})^{-1} \bmod p \quad (9)$$

Keterangan:

m = plaincitra
b = ciphercitra urutan genap
a = ciphercitra urutan ganjil
p = kunci publik
x = kunci privat

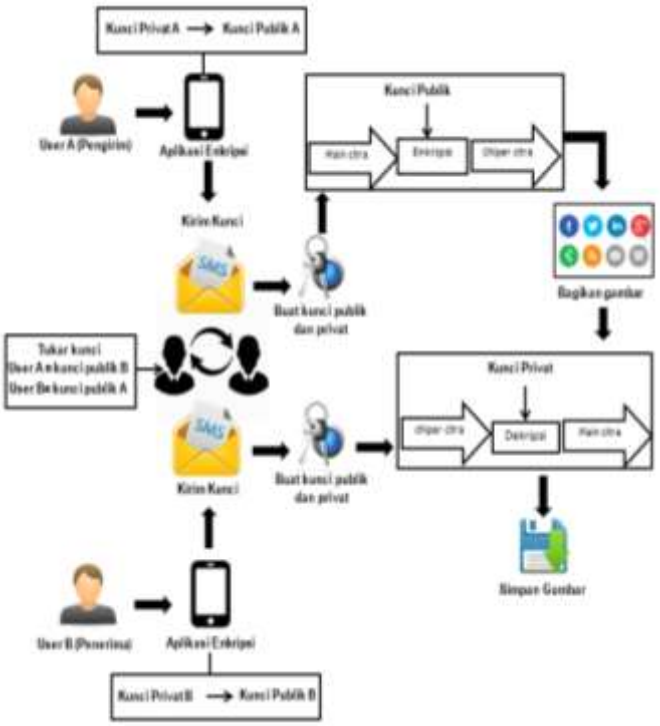
Langkah-langkah yang harus dilakukan penerima pesan (user B) adalah:

1. Memisahkan urutan dari pesan yang dikirim oleh user A, sesuai dengan ketentuan piksel yang sudah di set sebelumnya pada proses enkripsi. Pada pesan yang dikirim oleh si A yakni *a* = 20, dan *b*= 220.
2. Menghitung plain citra menggunakan persamaan (2.10) sehingga diperoleh $m = 20.220^{(331-1-26)} \pmod{331}$ adalah 192

Dari langkah ke-2 diatas, 192 merupakan piksel citra untuk itu si B yakni 192,192,192.

D. Perancangan Blok Diagram Sistem Secara Keseluruhan

Perancangan Blok diagram sistem secara keseluruhan digunakan untuk memperlihatkan gambaran yang jelas mengenai perancangan sistem yang akan dibuat. Perancangan sistem yang akan dibuat terdiri dari beberapa alur dimulai dari pembangkitan kunci rahasia, pembangkitan kunci publik untuk proses pertukaran kunci, proses pertukaran kunci, pembangkitan kunci untuk proses enkripsi dan dekripsi serta proses enkripsi dan dekripsi citra sidik jari.



Gambar 2 Rancangan Blok Diagram Sistem

Pada Gambar 2 merupakan aplikasi keseluruhan yang akan di rancang. Dalam aplikasi tersebut terdapat 2 *user*, salah satu menjadi pengirim dan yang lain akan menjadi penerima. Dalam hal ini sebelum adanya proses enkripsi dan dekripsi pengirim dan penerima harus membuat kunci. Pengirim dan penerima menginisialisasi suatu angka yang menjadi kunci rahasia (kunci privat). Selanjutnya pengirim dan penerima saling menghitung kunci publiknya masing-masing yang nantinya akan saling bertukar kunci publik. Proses pertukaran kunci tersebut dilakukan dengan proses pengiriman. Selanjutnya setelah pengirim dan penerima mendapat kunci publik barulah dihitung kunci privat yang digunkaan untuk proses dekripsi nantinya. Kunci privat yang didapat nantinya akan bernilai sama. Selanjutnya barulah pengirim melakukan proses enkripsi dengan menginputkan objek berupa citra sidik jari asli (plaincitra) yang akan menghasilkan citra sidik jari rahasia (chipercitra). Lalu pengirim mengirimkan *chipercitra* kepada penerima. Penerima akan melakukan proses dekripsi dengan menggunakan kunci privat sehingga menghasilkan citra sidik jari semula (plaincitra).

III. HASIL DAN PEMBAHASAN

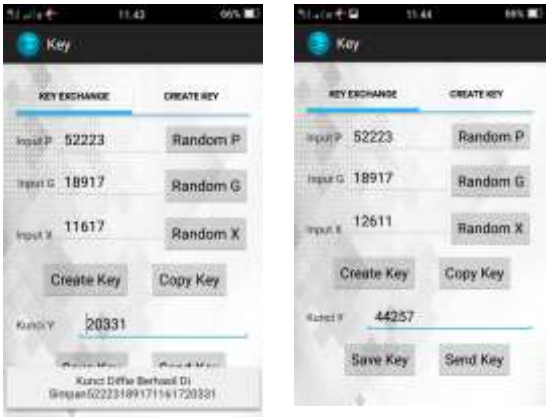
A. Pengujian Sistem

Pada tahap ini dilakukan pengujian terhadap aplikasi keamanan sidik jari menggunakan algoritma *Diffie Hellman* dan *Elgamal* pada *smartphone android* untuk mengetahui tingkat keberhasilan dari aplikasi yang telah dibuat dan memastikan aplikasi yang dibuat sesuai dengan perancangan. Adapaun pengujian yang dilakukan dalam pembuatan aplikasi enkripsi dan dekripsi sidik jari meliputi :

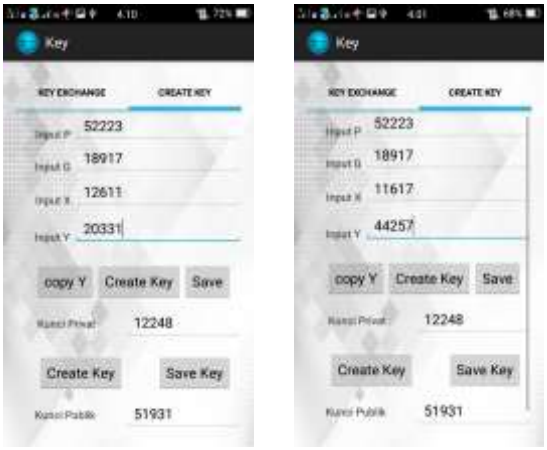
1. Pengujian proses pertukaran dan pembangkitan kunci.
2. Pengujian proses enkripsi.
3. Pengujian proses dekripsi.
4. Pengujian hasil citra.

B. Pengujian Proses Pertukaran, Pengiriman dan Penerimaan Kunci

Pengujian proses pertukaran kunci untuk mengetahui keberhasilan proses random kunci oleh pengirim, pembangkitan kunci *Y* oleh pengirim dan penerima, keberhasilan pertukaran kunci *Y* antara pengirim dan penerima, menyamakan penggunaan kunci *P* dan *G* antara pengirim dan penerima serta keberhasilan proses salin kunci dari pesan, baik salin kunci yang dilakukan oleh pengirim maupun penerima.



Gambar 3 Hasil Key Exchange Pengirim dan Penerima

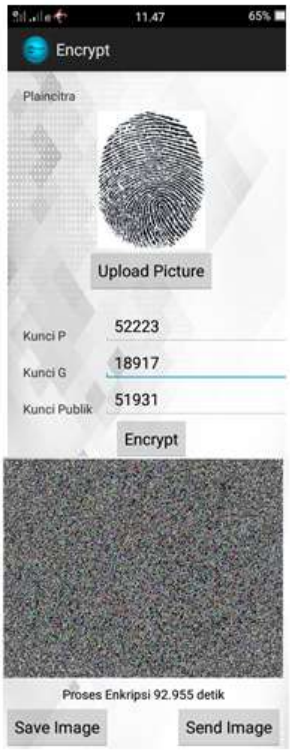


Gambar 4 Hasil Create Key Pengirim dan Penerima

Gambar 3 menunjukkan proses pembangkitan kunci *diffie* oleh masing-masing user, selanjutnya akan saling berbagi kunci Y, sehinnga didapat nilai kunci Y seperti yang ditunjukkan pada gambar 4 diatas. Selanjutnya pengirim dan penerima membangkitkan kunci public dan kunci privat. Dari hasil pembangkitan kunci didapatkan bahwa kunci privat pengirim = kunci privat penerima = 12248. Yang berarti proses pembangkitan kunci privat yang didapatkan sesuai dengan algoritma *Diffie Hellman*. Pada algoritma *Diffie hellman* kunci privat yang akan didapatkan oleh pengirim dan penerima bernilai sama, sehingga kunci publiknya juga akan sama sesuai ketentuan perhitungan kunci publik dengan menggunakan algorima *Elgamal*. Kunci public pengirim = kunci public penerima = 51931.

C. Pengujian Proses Enkripsi

Pengujian hasil enkripsi ditunjukkan pada gambar 17 di bawah ini:

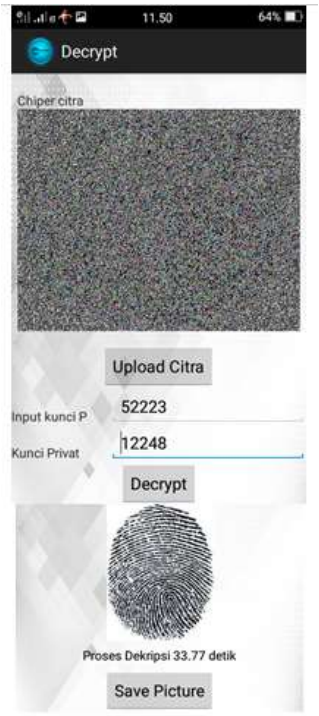


Gambar 5 Hasil Enkripsi

Gambar 5 menunjukkan Hasil enkripsi citra sidik jari yang menghasilkan *chiper* yang tidak dikenali objek aslinya sehingga gambar tidak dapat dimengerti, ukuran chiper yang dihasilkan berukuran 2 x ukuran gambar aslinya dikarenakan dalam algoritma *elgamal* menghasilkan 2 piksel *chiper* untuk setiap piksel yang di enkripsi. Gambar yang dihasilkan dapat disimpan kedalam 2 format yakni *.jpg dan *.png.

D. Pengujian Proses Dekripsi

Pengujian hasil enkripsi menunjukkan Hasil dekripsi chiper citra sidik jari yang diproses dengan algoritma *elgamal* agar menghasilkan *plain (*citra asli). Gambar yang dihasilkan dapat disimpan kedalam 2 format yakni *.jpg dan *.png. berikut tampilan hasil dekripsi sepertiyang ditunjukkan pada gambar 6 di bawah ini.



Gambar 6 Hasil Dekripsi

E. Pengujian Hasil Citra

Pengujian hasil citra untuk melihat keberhasilan enkripsi dekripsi citra yang akan di uji. Pengujian hasil citra yang akan diuji berdasarkan ukuran citra dan kunci yang digunakan..

1. Hasil Pengujian Berdasarkan Ukuran Citra

TABEL 1
HASIL PENGUJIAN CITRA BERDASARKAN UKURAN CITRA

NO	Citra Asli	Chiper	Citra output	Waktu	Keterangan
1	35 x 50 7,9 kb	100 x 70 20,8 kb	35 x 50 3,32 kb	Enkrip 2,88 detik Dekrip 1,87 detik	Citra berhasil dienkripsi dan didekripsi.
2	70 x 101 11,1 kb	202 x 140 71,8 kb	70 x 101 15,2 kb	Enkrip 15,66 detik Dekrip 9,64 detik	Citra berhasil dienkripsi dan didekripsi.
3	150 x 216 21,1 kb	432 x 300 316 kb	150 x 216 66 kb	Enkrip 74,19 detik Dekrip 27,14 detik	Citra berhasil dienkripsi dan didekripsi.

Dari Tabel 1 diatas menunjukkan bahwa dalam pengujian citra berdasarkan ukuran berhasil dilakukan, dilihat dari keberhasilan enkripsi dan dekripsi baik pada citra berukuran kecil maupun citra berukuran besar sekalipun. Citra yang berhasil dienkripsi dapat dilihat dari hasil enkripsi yang mana objek dari citra asli sudah tidak terlihat. Citra yang dihasilkan juga tidak menghasilkan citra berwarna hitam yang berarti setiap piksel chiper yang dihasilkan berhasil di konversi kedalam gambar kembali, dengan kata lain tidak ada nilai chiper yang melebihi dari nilai piksel rgb. Waktu enkripsi dan dekripsi dipengaruhi oleh ukuran, semakin besar dan bagus ukuran citra semakin lama proses enkripsi dan dekripsinya, serta sebaliknya.

2. Pengujian Berdasarkan Kunci

TABEL 2
HASIL PENGUJIAN CITRA BERDASARKAN KUNCI

Ukuran Kunci	Citra asli	Chiper citra	Plain citra	Keterangan
4 bit P=13 , g=5 , kunci public=5 , kunci privat=5		Enkrip 84,16	Dekrip 37,16	Proses enkripsi dan dekripsinya tidak berhasil.
8 bit P=229 , g=109 , kunci public=104 , kunci privat=32		Enkrip 82,77	Dekrip 31,12	Hasil enkripsi dan dekripsinya belum sepenuhnya berhasil, karena warna citra hasil dekripsi tidak sesuai.
16 bit P=63599 , g=31081 , kunci public=38085 , kunci privat=18986		Enkrip 92,35	Dekrip 33,45	Proses enkripsi dan dekripsi berhasil.

Dari Tabel 2 diatas menunjukkan bahwa dalam pengujian citra berdasarkan kunci untuk kunci bernilai 4 bit, citra gagal di enkripsi maupun didekripsi dikarenakan ukuran citra yang digunakan lebih besar daripada ukuran kunci yakni 8 bit, sehingga dalam proses perhitungannya akan dihasilkan nilai piksel rendah karena kunci yang digunakan sedikit sehingga piksel warna yang akan diset juga bernilai rendah yakni mendekati 0 sehingga citra yang dihasilkan dominan berwarna hitam.

Kunci 8 bit, citra berhasil di enkripsi karena objek yang dihasilkan tidak lagi menggambarkan citra aslinya, namun dalam proses konversi nilai rgb kedalam citra masih banyak terdapat piksel hitam, yang berarti masih ada nilai chiper yang rendah sehingga chiper akan diset bernilai 0 sehingga ada piksel yang hilang dari proses konversi ke citra rgb. hasil dekripsi terlihat dari objek citra aslinya, hanya saja untuk citra berwarna putih dengan piksel 255 tidak dapat diset dengan benar, dikarenakan piksel dari chiper untuk proses dekripsi tidak lagi sesuai karena ada nilai piksel rgb yang hilang.

Citra 16 bit, enkripsi dan dekripsinya berhasil dilakukan, terlihat bahwa hasil enkripsi dapat mengubah gambar asli menjadi gambar yang tidak dimengerti serta hasil dekripsi menunjukkan hasil citra sesuai dengan citra aslinya. Waktu yang digunakan untuk enkripsi dan dekripsi dipengaruhi oleh ukuran kunci, semakin besar ukuran kunci semakin lama waktu enkrispi dan dekripsinya, begitu pula sebaliknya.

IV . KESIMPULAN

Kesimpulan yang dapat di ambil dari penelitian yang sudah dilakukan adalah sebagai berikut:

1. Pertukaran kunci dengan menggunakan kunci *Diffie Hellman* dengan panjang kunci 16 bit dapat diimplementasikan dalam aplikasi android.
2. Proses enkripsi dan dekripsi citra sidik jari dengan menggunakan algoritma *Elgamal* dapat diimplementasikan kedalam aplikasi android.
3. Hal-hal yang mempengaruhi proses enkripsi dekripsi citra antara lain:
   - Ukuran gambar, semakin besar ukuran gambar semakin lama proses enkripsi
   - Ukuran kunci, semakin besar bit yang digunakan untuk pertukaran dan pembangitan kunci maka semakin lama proses enkripsi dan dekripsi.
4. Ukuran gambar hasil enkripsi menjadi dua kali lebih besar dari ukuran plain citra (citra asli) karena setiap piksel yang akan dienkripsi menghasilkan 2 piksel chiper.
5. Rata-rata waktu yang digunakan untuk enkripsi citra sidik jari yakni 102,482 detik dan waktu dekripsi berkisar 34,151 detik.
6. Waktu proses enkripsi lebih lama dibandingkan waktu dekripsi dikarenakan perhitungan matematika dalam proses enkripsi lebih banyak dibandingkan pada proses dekripsi.
7. Hardware yang digunakan untuk menjalankan aplikasi mempengaruhi kecepatan proses pertukaran kunci, pembangkitan kunci, enkripsi dan dekripsi.

REFERENSI

[1] Stallings, William. (2014). "Cryptography And Network Security Principles and Practice Sixth Edition". Pearson. USA

[2] Toni, 2016. *Rancang Bangun Aplikasi Kriptografi untuk Pengamanan Citra RGB 24 Bit dengan Menggunakan Algoritma Elgamal.* Tugas Akhir Mahasiswa Politeknik Negeri Lhokseumawe.

[3] Purwadi. 2014. *Aplikasi Kriptografi Asimetris dengan Metode Diffie Hellman dan Algoritma Elgamal untuk Kemanan Teks.* STMIK Triguna Dharma

[4] Stallings, W., 2005, *Cryptography and Network Security Principles and Practices Fourth Edition*, Prentice Hall, New Jersey

[5] Ingga. Michael, 2012. *Penggunaan Algoritma Diffie Hellman dalam melakukan pertukaran Kunci* . Sekolah Teknik elektro dan Informatika.

[6] Singh, R dan Kumar, S. 2012. "Elgamal's Algorihm in Criptography. International". *Journal of Scientific & Engineering Research* Volume 3, Issue 12.