Implementasi Algoritma Rc4 Untuk Enkripsi Dan Deskripsi Citra

Suryadi Hozeng, Sitti Aisa
STMIK Dipanegara
Jln. Perintis Kemerdekaan KM.9 Makassar, Telp. (0411) 587194 – Fax. (0411) 588284
e-mail : [*1]suryadi_hozeng@hotmail.com, [2] sittiaisa.11@gmail.com

Abstrak

Penggunaan citra digital yang tak terbatas sebagai koleksi pribadi serta, menjadi barang bukti yang mempunyai kekuatan hokum, karya seni fotografi yang juga mempunyai nilai jual tinggi dan lain – lain. Dimana, menuntut dibutuhkannya teknologi yang dapat menjaga keamanan citra pada saat didistribusikan. Untuk menghindari kemungkinan data yang disadap dapat langsung dibaca oleh penyadap, maka data yang dikirim diacak dengan menggunakan metode penyandian tertentu sehingga pesan yang terkandung dalam data yang dikirim tersebut menjadi lebih aman.Teknik kriptografi modern adalah transformasi matematika (algoritma) dengan menggunakan algoritma *RC4*. Teknik ini memperlakukan suatu pesan sebagai angka atau elemen aljabar dalam sebuah ruang dan mengubahnya menjadi sebuah pesan lain yang mempunyai arti, atau pesan yang dapat dimengerti (chipertext). Untuk menerjemahkan pesan – pesan tersebut, tentu harus ada proses yang mengembalikan informasi asli yang terkandung dalam pesan tersebut. Kita harus membalikan proses transformasi tersebut, proses pembalikan transformasi disebut juga dengan *dekripsi*. Biasanya algoritma enkripsi dan dekripsi diberikan parameter berupa *cryptographic keys* (kunci kriptografi), dengan implementasi algoritma ini dapat menjaga keamanan data citra yang akan didistribusikan.
.

Kata Kunci : Algoritma RC4, Citra Digital, Informasi

Abstract

The use of digital image infinite as well as private collections, as evidence that has legal force, the art of photography who also have high sales value and others - others. Where, demanding the need for technology that can maintain the security of the image at the time of distribution. To avoid the possibility of data being intercepted can be directly read by eavesdroppers, then the data is sent encrypted using an encryption method such that the messages contained in the data sent to more aman.Teknik modern cryptography is a mathematical transformation (algorithm) using the RC4 algorithm. This technique treats a message as numbers or algebra elements in a space and turn it into another message that has meaning, or message that can be understood (ciphertext). To translate the message - the message, there must be a process that restores the original information contained in the message. We must reverse the transformation process, the reversal of the transformation process is also called decryption. Usually the encryption and decryption algorithms are given parameters such as cryptographic keys (key cryptography), with the implementation of this algorithm can maintain the security of image data to be distributed.

Keywords: ***RC4 algorithm, Digital Image, Information***

1. PENDAHULUAN

Nilai guna data digital dalam hal ini citra, semakin luas dalam penggunaanya, semakin luas pengunaan citra digital yang tak terbatas sebagai koleksi pribadi tapi meluas dengan digunakannya citra digital sebagai barang bukti berkekuatan hukum, karya seni fotografi yang punya nilai jual tinggi dan lain-lain. menuntut dibutuhkannya sebuah teknologi yang dapat menjaga keamanan citra pada saat citra tersebut di distribusikan. Proses pengiriman data yang dilakukan media seperti *Local Area Network* (LAN), internet, email, handphone maupun media lain; pada dasarnya melakukan pengiriman data tanpa melakukan pengamanan terhadap konten dari data yang dikirim, sehingga ketika dilakukan penyadapan pada jalur pengirimannya maka data yang disadap dapat langsung dibaca oleh penyadap. Untuk menghidari kemungkinan data yang disadap dapat langsung dibaca oleh penyadap, maka data yang

dikirim diacak dengan menggunakan metode penyandian tertentu sehingga pesan yang terkandung dalam data yang dikirim tersebut menjadi lebih aman.

Teknik kriptografi modern adalah transformasi matematika (algoritma). Teknik ini memperlakukan suatu pesan sebagai angka atau elemen aljabar dalam sebuah ruang dan mengubahnya menjadi sebuah pesan lain yang mempunyai arti, atau pesan yang dapat dimengerti (chipertext). Untuk menerjemahkan pesan – pesan tersebut, tentu harus ada proses yang mengembalikan informasi asli yang terkandung dalam pesan tersebut. Kita harus membalikan proses transformasi tersebut, proses pembalikan transformasi disebut juga dengan *dekripsi*. Biasanya algoritma enkripsi dan dekripsi diberikan parameter berupa *cryptographic keys* (kunci kriptografi). Oleh karena itu, penulis meneliti untuk membangun suatu aplikasi pengamanan data citra dengan judul ***"Implementasi Algoritma RC4 Untuk Enkripsi Dan Dekripsi Citra"*** dengan harapan apalikasi ini dapat digunakan untuk menjaga keamanan data citra yang akan didistribusikan.

Tujuan penelitian ini adalah mengimplementasikan algortima kritografi RC4 untuk enkripsi dan dekripsi citra, serta membuat aplikasi yang dapat menjaga kerahasian citra dari pihak-pihak yang tidak berhak.

Kriptografi berasal dari bahasa Yunani yaitu *kryptos* yang berarti "rahasia" dan *graph* yang berarti "menulis" atau "belajar". Jadi kriptografi adalah ilmu dan seni yang mempelajari cara-cara untuk menyembunyikan informasi dengan cara menyamarkannya menjadi sandi yang sulit ditemukan maknanya [2]. Ada empat tujuan mendasari dari ilmu kriptografi ini yang juga merupakan aspek keamanan informasi, yaitu :

1. Kerahasiaan, adalah layanan yang digunakan untuk menjaga informasi dari pihak lain, yang dapat membuka informasi tersebut adalah pihak yang memiliki kunci rahasia untuk mendekripsi data tersebut.
2. Integritas data. Hal ini berhubungan dengan penjagaan dari perubahan data secara tidak sah. Untuk menjaga integritas data, sistem harus memiliki kemampuan untuk mendeteksi manipulasi data oleh pihak-pihak yang tidak mempunyai hak, antara lain penyisipan, penghapusan, dan penggantian data lain ke dalam data sebenarnya.
3. Autentikasi. Hal ini berhubungan dengan identifikasi/pengenalan, baik secara kesatuan sistem maupun informasi itu sendiri. Dua pihak yang saling berkomunikasi harus saling memperkenalkan diri. Informasi yang dikirimkan harus diautentikasi keasliannya, isi datanya, waktu pengiriman, dan lain lain.
4. Non Repudiasi, atau dapat disebut juga penyangkalan adalah usaha untuk mencegah terjadinya penyangkalan terhadap pengiriman/terciptanya suatu informasi oleh yang mengirimkan/membuatnya.

Algoritma yang berfungsi untuk melakukan tujuan kriptografi disebut sebagai algoritma sandi. Algoritma tersebut harus memiliki kekuatan untuk melakukan :

1. Konfusi/pembingungan, mempersulit pembaca biasa untuk memecahkan pesan yang sudah dienkripsi menjadi sandi-sandi tanpa memakai algoritma pendekripsinya.
2. Difusi/peleburan, yaitu dengan cara menghilangkan karakteristik dari informasi yang dienkripsi.

Jika algoritma sandi memiliki kekuatan tersebut maka algoritma tersebut dapat digunakan untuk mengamankan informasi. Pada implementasinya sebuah algoritma sandi harus memperhatikan kualitas layanan/QoS (*Quality of Service*) dari keseluruhan sistem dimana dia diimplementasikan. Algoritma sandi yang handal adalah algoritma sandi yang kekuatannya terletak pada kunci dan bukan kerahasiaan algoritma itu sendiri, maksudnya adalah kehandalan diukur dari tingkat kesulitan kunci pendekripsian dari sandi, bukan seberapa rahasia algoritma sandi tersebut. Teknik dan metode untuk menguji kehandalan algoritma sandi adalah kriptanalisa.[3]

Dasar matematis yang mendasari proses enkripsi dan dekripsi adalah relasi antar dua himpunan, dimana satu himpunan berisi elemen teks terang (*plaintext*) dan yang lain berisi elemen teks sandi (*ciphertext*).

Secara umum berdasarkan kesamaan kuncinya, algoritma sandi dibedakan menjadi :

1. Kunci Simetris/*symmetric-key*, sering disebut juga dengan kunci pribadi/ *private key.*
2. Kunci Asimetri/*asymmetric-key*, sering disebut juga dengan kunci publik/ *public key.*

Algoritma RC4 menghasilkan pseudorandom stream bit. Seperti halnya stream cipher lainnya, algoritma RC4 ini dapat digunakan untuk mengenkripsi dengan mengombinasikannya dengan plainteks dengan menggunakan bit-wise Xor (Exclusive-or) [1] Proses dekripsinya dilakukan dengan cara yang sama (karena Xor merupakan fungsi simetrik). Untuk menghasilkan keystream, cipher menggunakan state internal yang meliputi dua bagian :

1. Sebuah permutasi dari 256 kemungkinan byte.
2. Indeks-pointer 8-bit.

Permutasi di inisialisasi dengan sebuah variabel panjang kunci, biasanya antara 40 sampai 256 bit dengan menggunakan algoritma *key-scheduling* (KSA). Setelah proses ini selesai, stream yang terdiri dari sekumpulan bit tersebut terbentuk dengan menggunakan *Pseudo-Random Generation Algorithm* (PRGA). Berikut ini akan dijelaskan tentang kedua algoritma tersebut.

Tidak seperti stream cipher modern, RC4 tidak mengambil *nonce yang* terpisah bersamaan dengan kunci[1]. Hal ini berarti jika kunci *single long-term* digunakan untuk mengenkripsi beberapa stream, kriptosistemnya harus menentukan bagaimana cara mengombinasikan *nonce* tersebut dan kunci *long-term* untuk menghasilkan kunci stream untuk RC4. Sebuah pendekatan untuk menangani hal tersebut adalah dengan membuat sebuah kunci RC4 dengan menggunakan fungsi *hash*. Enkripsi dengan menggunakan RC4 dapat diterobos dan rentan terhadap *bit-flipping attack.* Untuk menanggulangi hal ini, skema enkripsi harus dikombinasikan dengan *message authentication code* yang kuat. Research Map Algortima RC4 di Indonesia adalah :

1. "Aplikasi Sistem Pengaman Data Dengan Metode Enkripsi Menggunakan Algoritma RC4" ditulis oleh Siti Mariyam di Universitas Narotama Surabaya. Pada penelitian ini penulis membangun sebuah aplikasi desktop dengan menerapkan RC4 untuk mengenkrispi file teks. File yang dapat dieksekusipun sebatas file dengan ekstensi txt.
2. Implementasi Algoritma RC4 Untuk Keamanan Login Pada Sistem Pembayaran Uang Sekolah (Studi Kasus : Di Yayasan Yabis Bontang). Ditulis oleh Ansar Rizal dan Suharto pada tahun 2011. Pada penelitian ini peneliti mencoba melakukan enkripsi ganda, dengan melakukan enkripsi ulang terhadap cipher teks yang telah dienkripsi.
3. Pembangunan *Message Authentication Code MAC* Berbasis *Cipher* Aliran (RC4). Ditulis oleh Made Harta Dwijaksara di Institute Teknologi Bandung pada penelitian ini penulis membangun sebuah aplikasi MAC untuk menjaga ontetikasi pesan menggunakan algoritma RC4.

2. Metode Penelitian

2.1 Alat dan Bahan

Pada penelitian ini penulis menggunakan alat bantu dalam menganalisis dan mempelajari sistem yang ada dan sistem yang akan dirancang.

Alat desain penelitian, terdiri atas :

a. Use Case Diagram
b. Class Diagram
c. Sequence Diagram
d. Activity Diagram

Adapun perangkat keras yang digunakan yaitu sebuah laptop dengan spesifikasi :

a. Processor : Intel® Core 2 Duo 2.6 GHz
b. Memory (RAM) : 1 GB.
c. Harddisk : 320 GB.

Perangkat lunak yang digunakan yaitu :

a. Sistem Operasi Microsoft Windows 7
b. Bahasa Pemrograman (PHP ver 5.0.1, MySQL ver 5.0.15).
c. App Serv
d. Adobe Dreamweaver 8.0
e. Browser (Mozilla FireFox 4.0) .

2.2 Metode Pengujian

Metode pengujian yang digunakan dalam penelitian ini adalah metode pengujian perangkat lunak *Black Box.* Metode ujicoba *blackbox* memfokuskan pada keperluan fungsional dari *software*, Karna itu ujicoba *blackbox* memungkinkan pengembang *software* untuk membuat himpunan kondisi input yang akan melatih seluruh syarat-syarat fungsional suatu program.

Ujicoba blackbox berusaha untuk menemukan kesalahan dalam beberapa kategori, diantaranya :

1. Fungsi-fungsi yang salah atau hilang
2. Kesalahan interface
3. Kesalahan dalam struktur data atau akses database eksternal
4. Kesalahan performa
5. Kesalahan inisialisasi dan terminasi

2.3 Tahap – Tahap Penelitian

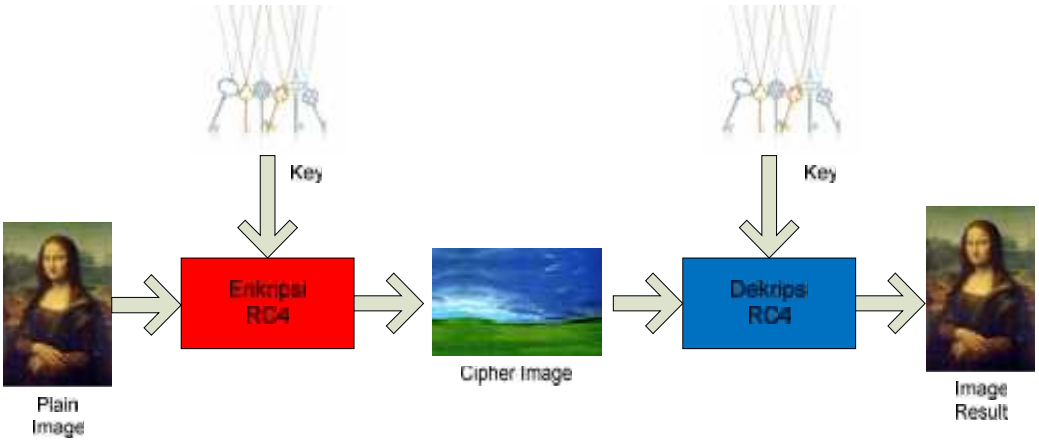
Langkah-langkah yang dilakukan dalam penelitian adalah:

1. Pengumpulan data
   Mengunpulkan data – data dan informasi untuk di jadikan acuan dalam membangun aolikasi yang dirancang.
2. Analisis algoritma
   Mengidentifikasi dan mengenali masalah yang ada, kemudian mencari alternative – alternative pemecahannya.
3. Desain algortima
   Setelah masalah ditentukan dan dianalisa data sudah dilakukan maka perlu di lakukan desain algoritma sesuai dengan masalah yang dihadapi dalam hal ini enkripsi dan dekrispsi citra
4. Implemtasi
   Mengimplementasikan hasil desain algoritma untuk mengenkripsi dan mendekripsi citra.
5. Pengujian aplikasi
   Setelah proses analisa, desain dan implementasi selesai,dilakukan pengujian sistem menggunakan metode *Black box.*

3. Hasil dan Analisis

3.1 Desain Aplikasi

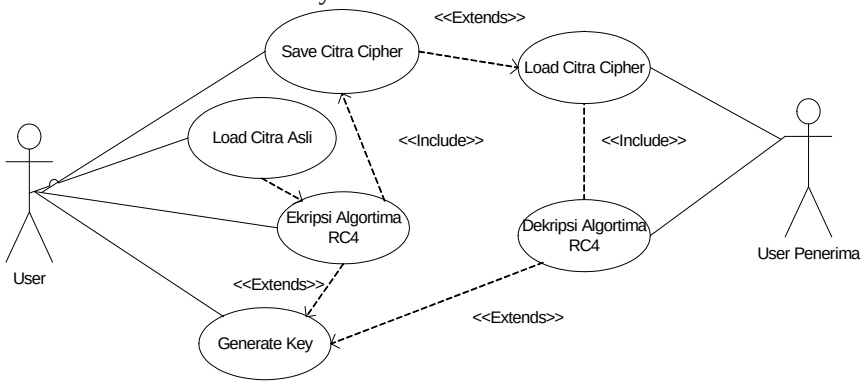
Langkah awal perancangan dalam pembuatan aplikasi enkripsi gambar dengan algortima RC4 membuat dokumentasi desain aplikasi dengan menggunakan UML (*Unfield Modelling nguage*), dengan menggunakan beberapa buah diagram, yaitu : *use case diagram*, kemudian membuat *class diagram*, *sequence diagram* dan yang terakhir *activity diagram* yang menunjukkan setiap aktivitas pada aplikasi enkripsi gambar ini.



Gambar 3.1. 1 : Desain Aplikasi

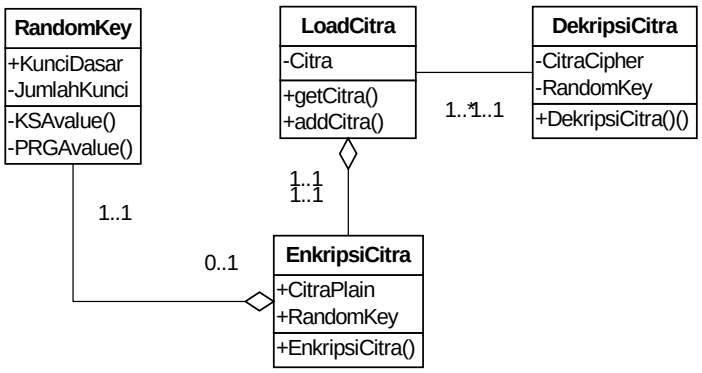
3.2 Use Case Diagram

Tingkah laku aplikasi di defenisikan pada diagram use case di atas, terlihat bagiamana user sebagai faktor pemicu dari aplikasi dengan menginput citra yang akan dienkripsi hasil ekripsi gambar berupa citra cipher, citra cipher ini yang akan menjadi input pada fungsi dekripsi untuk mengembalikan citra kebentuk awalnya.



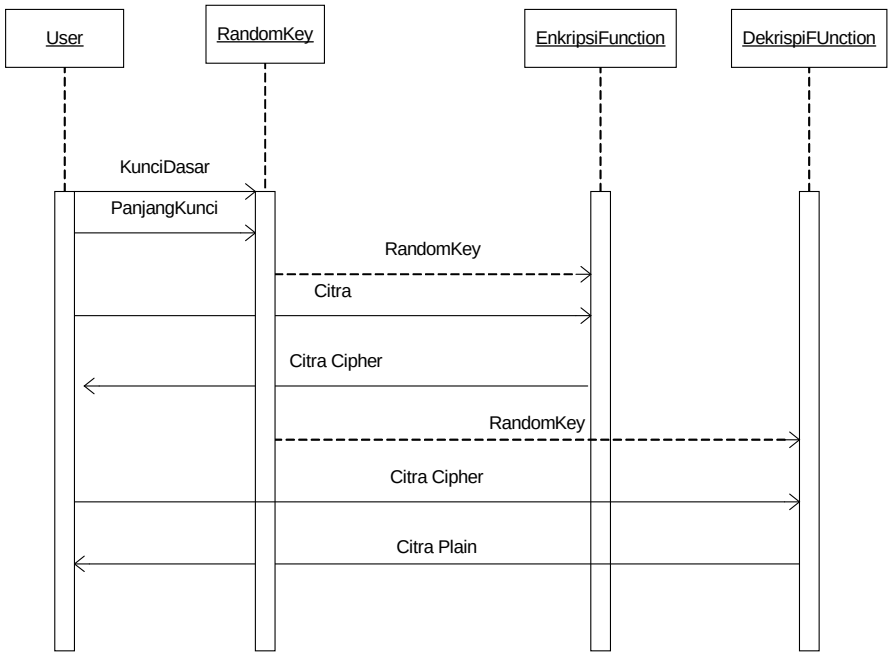
Gambar 3.2.1 : Use Case Diagram

3.3 Class Diagram



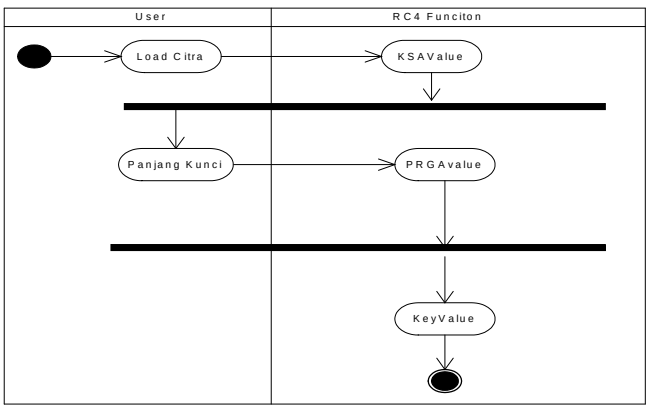
Gambar 3.3 1 : Class Diagram

3.4 Sequence Diagram

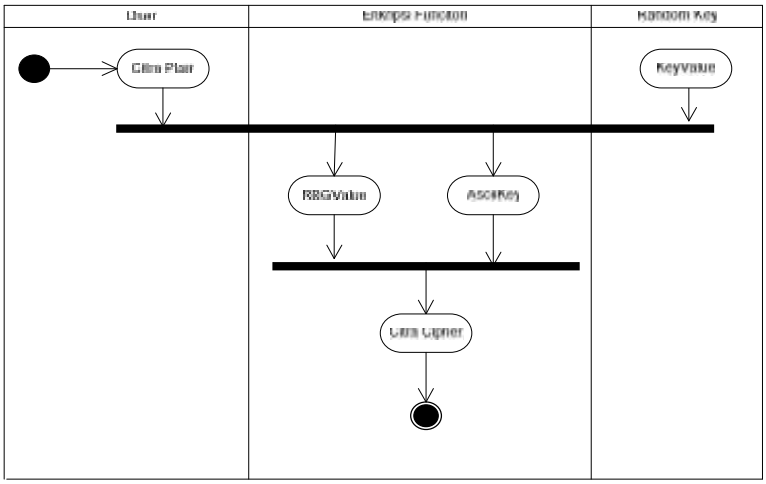


Gambar 3.4 1 : Sequence Diagram

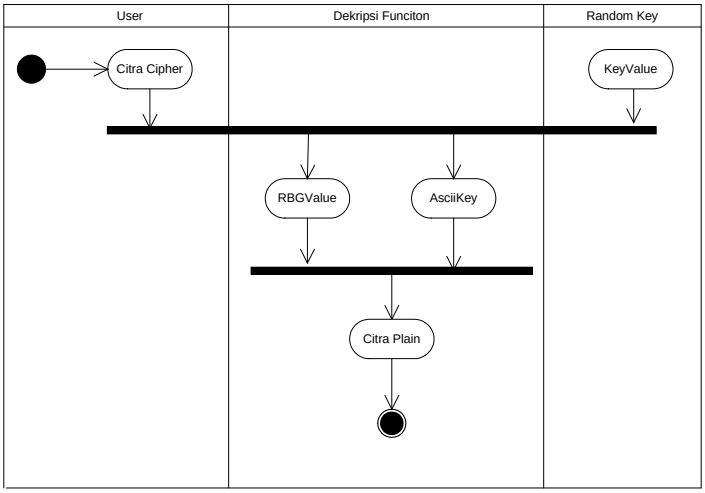
3.5 Activity Diagram



Gambar 3.5 1 : Activity Diagram Random Key RC4



Gambar 3.5 2 : Activity Diagram Enkripsi Citra



Gambar 3.5 3 : Activity Diagram Deskripsi Citra

3.6 Hasil Implementasi Algoritma RC4

1. Pembangkit Kunci RC4

RC4 menghasilkan pseudorandom stream bit. Seperti halnya stream cipher lainnya, algoritma RC4 ini dapat digunakan untuk mengenkripsi dengan mengombinasikannya dengan plainteks dengan menggunakan bit-wise Xor (Exclusive-or). Proses dekripsinya dilakukan dengan cara yang sama (karena Xor merupakan fungsi simetrik).

Permutasi di inisialisasi dengan sebuah variabel panjang kunci, dengan menggunakan algoritma *key-scheduling* (KSA). Setelah proses ini selesai, stream yang terdiri dari sekumpulan bit tersebut terbentuk dengan menggunakan *Pseudo-Random Generation Algorithm* (PRGA).

a. key-scheduling (KSA)

Algoritma key scheduling digunakan untuk menginisialisasi permutasi di array “S”. panjang kunci didefinisikan sebagai jumlah byte di kunci dan mempunyai rentang panjang kunci dari 1 sampai 256, khususnya antara 5-16 tergantung dari panjang kunci 40128bit. Pertama-tama array “S” diinisialisasi untuk identitas permutasi. S kemudian diproses ke 256 iterasi dengan cara yang sama dengan PRGA utama, tapi juga dikombinasikan dalam byte dari kunci dalam waktu yang bersamaan. Berikut Pseudocode algoritma KSA :

```
for i from 0 to 255
  S[i] := i
```

```
Endfor
        j := 0
    for i from 0 to 255
    swap values of S[i] and S[j]
   endfor
```

dan berikut implementasi adalah algoritma KSA dalam Javascript :

```
function  KSA(key_length) {
        for (i = 0; i < 256; i++){
       S[i] = i;
  }
        for (i = j = 0; i < 256; i++) {
     j = (j + key[i % key_length] + S[i]) & 255;
            swap(i, j);        }
     i = j = 0;
  }
```

b. Pseudo-Random Generation Algoritma

PRGA (*Pseudo-Random Generation Algoritma*) memodifikasi state dan output sebuah byte dari keystream. Hal ini penting karena banyaknya dibutuhkan iterasi. Dalam setiap iterasi, PRGA menginkremen **i**, menambahkan nilai S yang ditunjuk oleh **I** sampai **j,** kemudian menukar nilai S[i] dan S[j], lalu mengembalikan elemen dari S di lokasi S[i] + S[j] (modulo 256). Setiap elemen S ditukar dengan elemen lainnya paling tidak satu kali setiap 256 iterasi.

Pseudocode algoritma PRGA dapat dilihat sebagai berikut :

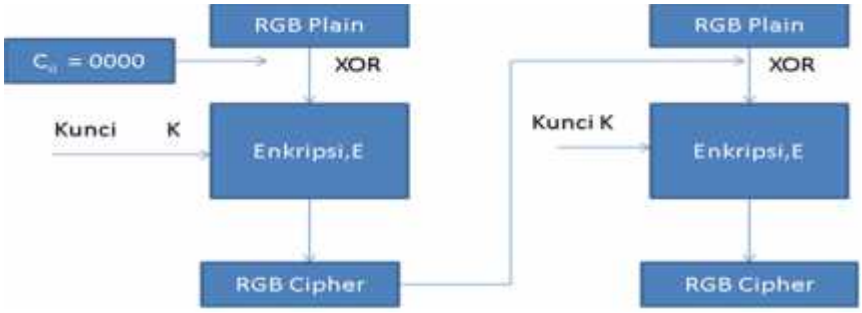
```
i := 0 ; j := 0
while GeneratingOutput:
        i := (i + 1) mod 256
 j := (j + S[i]) mod 256
        swap values of S[i] and S[j]
        K := S[(S[i] + S[j]) mod 256]
        output K
endwhile
```

Implementasi Algortima PRGA

```
function PRGA( ) {
        i = (i + 1) & 256;
        j = (j + S[i]) & 256;
        swap( i, j);
        return S[(S[i] + S[j]) & 1256];
}
```

1. Enkripsi Citra

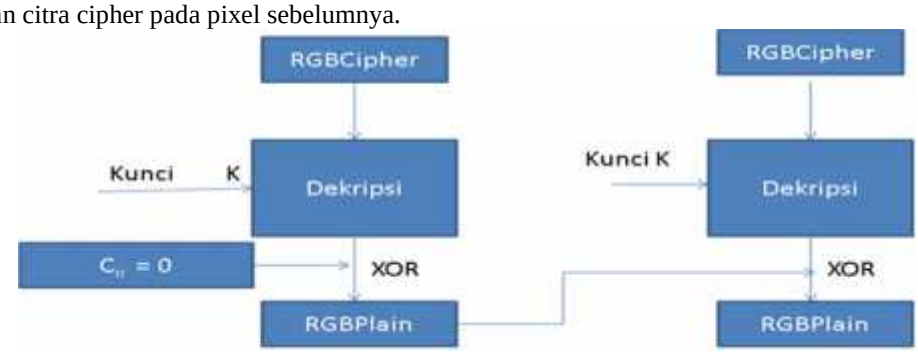
Metode enkripsi yang digunakan menggunakan skema *Cipher Blok Chainin*g dimana setiap stream sebelum tiap citra di enkripsi dengan kunci terlebih dahulu pixel citra di Xor dengan citra cipher pada pixel sebelumnya.



Gambar 3.6.2 : Skema enkripsi tiap pixel di dalam citra

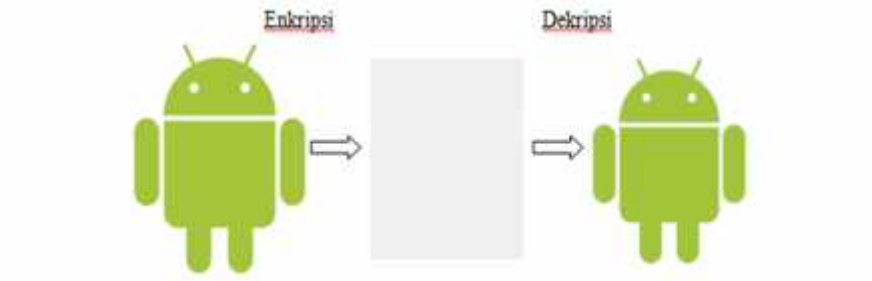
2. Dekripsi Citra

Metode dekripsi yang digunakan menggunakan skema *Cipher Blok Chainin*g dimana setiap stream citra di enkripsi dengan kunci kemudian di Xor dengan terlebih dahulu pixel citra di Xor dengan citra cipher pada pixel sebelumnya.



Gambar 3.6.3 : Skema Dekripsi tiap pixel di dalam citra

3.7 Uji Coba Enkripsi Citra



Gambar 3.7. 1 : Hasil Ujicoba Citra Android.JPG

3.8 Metode Pengujian Black Box

Sebelum penerapan sistem, terlebih dahulu sistem harus bebas dari kesalahan yang mungkin dapat terjadi. Sistem dalam hal ini program di uji untuk tiap tiap modul yang ada dan dilanjutkan dengan pengujian untuk semua modul yang telah dirangkai. Pengujian program yang dilakukan dengan menggunakan metode pengujian *white box* dan *basis path.* Serta pengujian fungsional dari aplikasi tanpa memperhatikan proses-prose yang terjadi di dalamnya yaitu pengujian *black box*

Metode ujicoba *blackbox* memfokuskan pada keperluan fungsional dari *software*, Karna itu ujicoba *blackbox* memungkinkan pengembang *software* untuk membuat himpunan kondisi input yang akan melatih seluruh syarat-syarat fungsional suatu program.

Ujicoba blackbox berusaha untuk menemukan kesalahan dalam beberapa kategori, diantaranya :

1. Fungsi-fungsi yang salah atau hilang
2. Kesalahan interface
3. Kesalahan dalam struktur data atau akses database eksternal

4. Kesalahan performa
5. Kesalahan inisialisasi dan terminasi

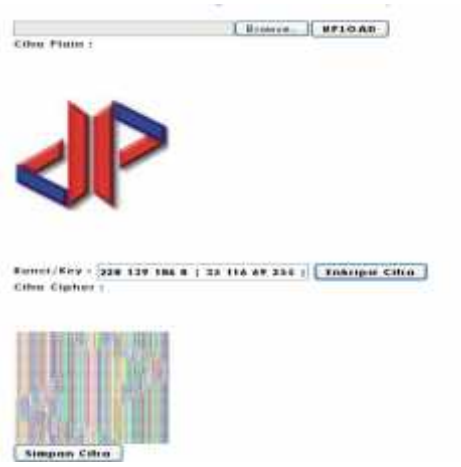
a. Fungsi Pembangkit Kunci Enkripsi

Tabel 3.8 1 Pengujian Pembangkit Kunci Enkripsi

Test Factor	Hasil	Keterangan
Menampilkan kunci dari proses input kunci dasar dan panjang kunci	✓	Modul dapat menghasilkna kunci berupa karakter dan nilai desimal
Antarmuka		
		

b. Fungsi Enkripsi

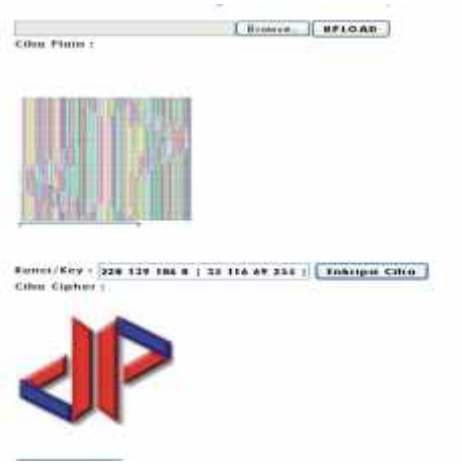
Tabel 3.8 .2 Pengujian Fungsi Enkripsi

Test Factor	Hasil	Keterangan
Dapat memangil citra yang akan di enkripsi kemuadian menghasilkan citra cipher	✓	Modul dapat menghasilkan citra berupa citra cipher
Antarmuka		
		

c. Fungsi Dekripsi

Tabel 3.8.3 Pengujian Fungsi Dekripsi

Test Factor	Hasil	Keterangan
Dapat memangil citra cipher yang akan di dekkripsi kemuadian menghasilkan citra asli	✓	Modul dapat menghasilkan citra berupa citra asli

Antarmuka


4. Kesimpulan

Berdasarkan hasil perancangan Aplikasi enkripsi citra dengan algortima RC4 berbasis PHP maka dapat ditarik kesimpulan sebagai berikut:

1. Aplikasi ini dapat digunakan untuk mengenkripsi citra guna kebutuhan kerahasiaan citra dalam proses pengirim citra.
2. Aplikasi ini dapat digunakan untuk mengdekripsi citra guna mengembalikan citra pada gambar aslinya untuk digunakan sebagai mana mestinya
3. Penerapan Aplikasi berbasis PHP memudahkan seluruh cabang perusahaan untuk menggunaknnya secara bersama-sama demi keamanan gambar yang aka dikirimkan.

DAFTAR PUSTAKA

[1] Arie Pratama Sutiono *"Algoritma RC4 sebagai Perkembangan Metode Kriptografi "*SekolahTeknik Elektro dan Informatika Institut Teknologi Bandung, 2011

[2] Rinaldi Munir .,"*Kriptografi"* Informatika, Bandung 2006

[3] Sutoyo "*Pengelolaan Citra Digital"*, Andi Offset, Yogyakarta 2009.