ISSN: 2461-1409

IMPLEMENTASI KRIPTOGRAFI ALGORITMA AES SERTA ALGORITMA KOMPRESI HUFFMAN DENGAN MENGGUNAKAN PEMROGRAMAN PHP

Sugeng Santoso[1]
Sutrisno[2]
Gusri Hardiyanto[3]

Alumni Magister Ilmu Komputer Universitas Budi Luhur Jakarta [1, 2], STMIK Raharja Jurusan Teknik Informastika[3]
Jl. Jendral Sudirman No. 40, Modernland, Tangerang
Email: sugeng.santoso@raharja.info, kmnpusat@gmail.com, gusri.hardiyanto@gmail.com

ABSTRAK

Perkembangan ilmu pengetahuan di segala bidang begitu pesat, terutama dalam bidang teknologi informasi dan komputerisasi yang semakin maju dan modern. Khusus perkembangan ilmu Teknik Informatika pada masa sekarang ini juga sangatlah cepat dan maju, semakin meningkatnya persaingan menjadikan dunia perusahaan atau instansi berlomba menggunakan sistem yang terbaik untuk mencapai kesuksesan yang pesat. Perkembangan yang sangat signifikan menjadikan segala kegiatan komputerisasi yang terhubung melalui jaringan internet menjadi sangat beresiko terhadap pencurian data. Perkembangan teknologi mendorong masyarakat untuk dapat mengikuti kemajuan teknologi yang sedang berkembang. Dengan akses yang mudah, fasilitas yang cukup banyak, masyarakat dapat belajar teknologi informatika. Dengan demikian memunculkan sisi positif dan negatif dari hasil pembelajaran ilmu yang dilakukan. Meninjau dari sisi negatif, kasus yang sering terjadi adalah pencurian data penting milik seseorang, perusahaan, ataupun instansi pemerintahan. Oleh karena banyaknya kasus pencurian data yang terjadi, para ahli di bidang informatika melakukan tindakan untuk mencari solusi agar data-data yang bersifat penting dapat dilindungi dengan baik, sekalipun data tersebut berhasil dicuri oleh pihak yang tidak bertanggung jawab. Oleh karena itu, penulis mencoba membuat sebuah aplikasi sederhana untuk dapat melindungi data-data penting.

***Kata kunci** : Aplikasi, Kriptografi, Keamanan, Algoritma.*

ABSTRACT

Development of science in all fields so rapidly, especially in the field of information technology and computerization are more advanced and modern. Specifically the development of Information Engineering at the present time is also very quick and forward, increasing competition makes the world competing company or agency using the best system to achieve rapid success. A very significant development makes all the computerized activities that are connected through the Internet to be very risky against data theft. The development of technology to encourage people to be able to follow the progress of emerging technologies. With easy access, the facilities pretty much, people can learn information technology. Thus bring positive and negative sides of the learning outcomes of science are done. Reviewing the negative side, frequent case is the theft of important data belonging to a person, company, or government agencies. Therefore, many cases of data theft happens, experts in the field of informatics to take action to find a solution to the data that is important can be protected very well, even if the data is

stolen by irresponsible parties. Therefore, the author tries to make a simple application to be able to protect important data.

***Keywords**: Application, Cryptography, Security, Algorithm.*

PENDAHULUAN

Perkembangan teknologi informatika yang sangat cepat dan pesat, membawa perubahan besar disegala bidang. Salah satu diantaranya adalah perekonomian. Perkembangan teknologi membangkitkan bidang usaha dengan sangat cepat, setiap kegiatan usaha dapat di topang dengan adanya teknologi yang maju. Perkembangan teknologi menjadikan ketatnya persaingan di dunia usaha.

Dengan memanfaatkan teknologi setiap pelaku usaha dapat dengan mudah berkomunikasi, bertukar data-data penting dalam usaha tidak perlu lagi harus bertatap muka bertemu langsung. Karena dengan memanfaatkan teknologi semua bisa dilakukan dari jarak yang jauh sekalipun. Teknologi memberikan profit yang besar dalam membantu kegiatan usaha mencapai kesuksesan dengan sangat cepat.

Tapi selain memberikan keuntungan, teknologi juga memberikan dampak kerugian yang cukup besar. Dengan melakukan pertukaran data-data penting dalam kegiatan usaha, resiko data yang dimiliki untuk dicuri oleh pihak luar masih sangat besar. Karena hal tersebut maka sangat dibutuhkan sebuah cara untuk mengamankan data-data yang dikirimkan melalui sistem komputer.

Untuk mengamankan data-data yang penting dapat dilakukan dengan cara enkripsi (*encryption*) atau sering disebut dengan Kriptografi. Tujuan dari enkripsi/ kriptografi adalah membuat data-data penting sulit untuk dibaca sekalipun data tersebut berhasil dicuri pihak yang tidak bertanggung jawab. Dalam kriptografi terdiri dari dua hal, enkripsi (*encryption*) yaitu proses merubah data asli *(plain text*) menjadi data samaran (*chiper text*) dan dekripsi (*decryption*) yaitu proses pengembalian *chiper text* menjadi *plain text* kembali.

LANDASAN TEORI

Kriptografi

Kriptografi berasal dari bahasa Yunani : “cryptos” artinya “secret” (rahasia), sedangkan “graphein” artinya “writing” (tulisan). Jadi, kriptografi berarti “secret writing” (tulisan rahasia).

“Kriptografi adalah ilmu yang mempelajari teknik-teknik matematika yang berhubungan dengan aspek keamanan informasi seperti kerahasiaan, integritasdata, serta otentikasi” (Alfred J. Menezes,2001).

2. Algoritma Advanced Encryption Standard (AES)

Menurut Rinaldi Munir (2006), algoritma AES menggunakan substitusi dan permutasi, dan sejumlah putaran (chiper berulang), dimana setiap putaran menggunakan kunci yang berbeda (kunci setiap putaran disebut round key).

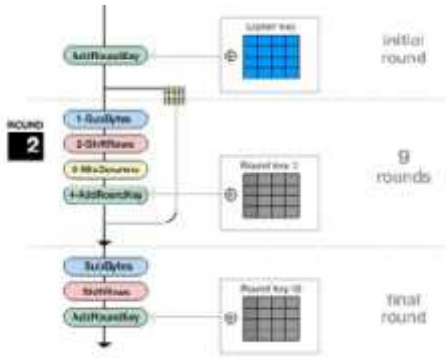
AES menetapkan panjang kunci 128, 192, dan 256 bit. Karena itu, maka dikenal AES-128, AES-192, dan AES-256.

Tabel 1 : Perbedaan versi AES

Nb: 1 word = 32 bit

	Panjang Kunci (Nk words)	Ukuran Blok (Nb words)	Jumlah Putaran (Nr)
AES 128	**4**	**4**	**10**
AES 192	**6**	**4**	**12**
AES 256	**8**	**4**	**14**

Proses Enkripsi Algoritma AES



Gambar 1 : Diagram Proses Enkripsi AES

Garis besar Algoritma AES Rijndael yang beroprasi pada blok 128-bit, dengan kunci 128-bit adalah sebagai berikut (di luar proses pembangkitan *round key).*

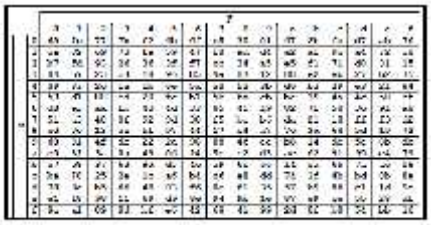
a. *AddroundKey*: Melakukan XOR antara *state* awal (*plaintext*) dengan *chiperkey*. Tahapan ini sering disebut juga *initial round*.

b. *Round* : Putaran sebanyak Nr-1 kali. Proses yang dilakukan pada setiap putaran adalah:

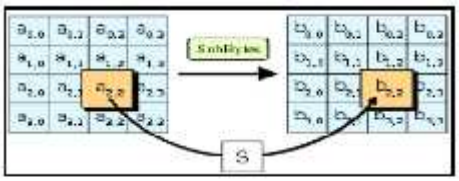
1) *SubBytes* : Putaran *byte* dengan menggunakan table substitusi.

   Proses *SubBytes* memetakan setiap *byte* dari *array State* dengan menggunakan tabel substitusi S-Box. Tidak seperti Des S-box berbeda pada setiap putaran, AES hanya mempunyai satu buah S-Box

Tabel 2 S-Box Sub Bytes



(Sumber: *Federal Information Processing Standards Publication 197*, 2001, h. 16)

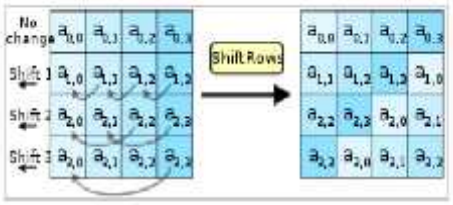


(Sumber: Munir, 2006, h.163)

Gambar 2 : Transformasi SubBytes

2) *ShiftaRows*: Pergeseran baris-baris *array state* secara *wrapping*.

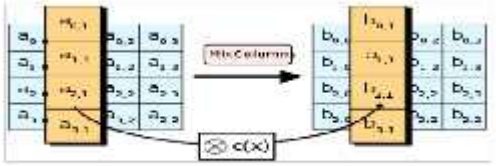
   Melakukan pergeseran secara *wrapping* (siklik) pada baris terakhir dari array state.Jumlah pergeseran bergantung pada nilai baris (r). Baris r=1 digeser sejauh 1 *byte*, baris r=2 digeser sejauh 2 *byte* Dan baris r=3 digeser sejauh 3 *byte*. Baris r=0 tidak digeser



(Sumber: Munir, 2006, h. 165)

Gambar 3 :Transformasi Shift Row

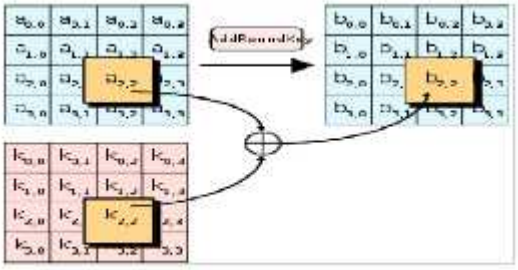
3) *MixColumn* : Mengacak data di masing-masingkolom array state.



(Sumber: Munir, 2006, h. 166)

Gambar 4 : Transformasi Mix Columns

4) *addroundKey* : Melakukan XOR antara state sekarang *roundKey*.



(Sumber: Munir, 2006, h. 166)

Gambar 5 : Transformasi addRoundKey

c. *Final Round* : Proses untuk putaran terakhir.
   1. *subBytes*
   2. *shiftRows*
   3. *addroundKey*

3. Algoritma Kompresi Huffman

Algoritma Huffman, yang dibuat oleh seorang mahasiswa MIT bernama David Huffman pada tahun 1952, merupakan salah satu metode paling lama dan paling terkenal dalam kompresi teks .

Algoritma Huffman menggunakan prinsip pengkodean yang mirip dengan kode Morse, yaitu tiap karakter (simbol) dikodekan hanya dengan rangkaian beberapa bit, dimana karakter yang sering muncul dikodekan dengan rangkaian bit yang pendek dan karakter yang jarang muncul dikodekan.dengan rangkaian bit yang lebih panjang.

Berdasarkan tipe peta kode yang digunakan untuk mengubah pesan awal (isi data yang diinputkan) menjadi sekumpulan codeword, algoritma Huffman termasuk kedalam kelas algoritma yang menggunakan metode statik .

Metoda statik adalah metoda yang selalu menggunakan peta kode yang sama, metoda ini membutuhkan dua fase (two-pass): fase pertama untuk menghitung probabilitas kemunculan tiap simbol dan menentukan peta kodenya, dan fase kedua untuk mengubah pesan menjadi kumpulan kode yang akan di taransmisikan.

Sedangkan berdasarkan teknik pengkodean simbol yang digunakan, algoritma Huffman menggunakan metode symbolwise. Metoda symbolwise adalah metode yang menghitung peluang kemunculan dari setiap simbol dalam satu waktu, dimana simbol yang lebih sering muncul diberi kode lebih pendek dibandingkan simbol yang jarang muncul.

Proses Encoding

Encoding adalah cara menyusun string biner dari teks yang ada. Proses encoding untuk satu karakter dimulai dengan membuat pohon Huffman terlebih dahulu. Setelah itu, kode untuk satu karakter dibuat dengan menyusun nama string biner yang dibaca dari akar sampai ke daun pohon Huffman. Langkah-langkah untuk men-encoding suatu string biner adalah sebagai berikut :

a. Tentukan karakter yang akan di-encoding
b. Mulai dari akar pohon Huffman, baca setiap bit yang ada pada cabang yang bersesuaian sampai

ketemu daun dimana karakter itu berada.

c. Ulangi langkah-langkah tersebut sampai seluruh karakter diencoding.

Tabel 3 : Kode Huffman

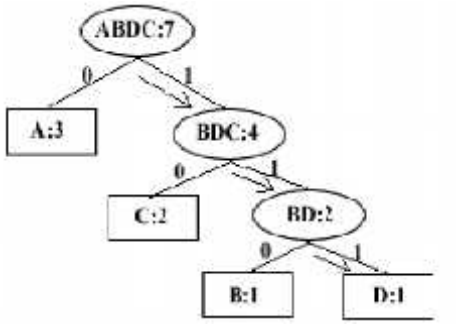
Karakter	String Biner Huffman
A	0
B	110
C	10
D	111

Proses Decoding

Decoding merupakan kebalikan dari encoding. Decoding berarti menyusun kembali data dari string biner menjadi sebuah karakter kembali. Decoding dapat dilakukan dengan dua cara, yang pertama dengan menggunakan pohon Huffman dan yang kedua dengan menggunakan tabel kode Huffman. Langkah-langkah men - decoding suatu string biner dengan menggunakan pohon Huffman adalah sebagai berikut :

a. Baca sebuah bit dari string biner.
b. Mulai dari akar
c. Untuk setiap bit pada langkah 1, lakukan traversal pada cabang yang bersesuaian.
d. Ulangi langkah 1, 2 dan 3 sampai bertemu daun. Kodekan rangkaian bit yang telah dibaca dengan karakter di daun.

Ulangi dari langkah 1 sampai semua bit di dalam string habis. Sebagai contoh kita akan men-decoding string biner yang bernilai ”111”



Gambar 6 : Pohon Huffman

Setelah kita telusuri dari akar, maka kita akan menemukan bahwa string yang mempunyai kode Huffman “111” adalah karakter D.

Cara yang kedua adalah dengan menggunakan tabel kode Huffman. Sebagai contoh kita akan menggunakan kode Huffman pada Tabel 3 untuk merepresentasikan string “ABACCDA”. Dengan menggunakan Tabel 3 string tersebut akan direpresentasikan menjadi rangkaian bit : 0 110 0 10 10 1110.

Jadi, jumlah bit yang dibutuhkan hanya 13 bit. Dari Tabel 3 tampak bahwa kode untuk sebuah simbol/karakter tidak boleh menjadi awalan dari kode simbol yang lain guna menghindari keraguan (ambiguitas) dalam proses dekompresi atau decoding. Karena tiap kode Huffman yang dihasilkan unik, maka proses decoding dapat dilakukan dengan mudah.

Contoh: saat membaca kode bit pertama dalam rangkaian bit “011001010110”, yaitu bit “0”, dapat langsung disimpulkan bahwa kode bit “0” merupakan pemetaan dari simbol “A”. Kemudian baca kode bit selanjutnya, yaitu bit “1”. Tidak ada kode Huffman “1”, lalu baca kode bit selanjutnya, sehingga menjadi “11”. Tidak ada juga kode Huffman “11”, lalu baca lagi kode bit berikutnya, sehingga

menjadi “110”. Rangkaian kode bit “110” adalah pemetaan dari simbol “B”.

RATIONAL UNIFIED PROSES (RUP)

Rational Unified Proses (RUP) memiliki empat tahap, yaitu:

a. Tahap Inception (Tahap Analisis)

Pada tahap ini pengembang mendefinisikan batasan kegiatan, melakukan analisis kebutuhan user, dan melakukan perancangan awal perangkat lunak (perancangan arsitektural dan use case).

b. Tahap Elaboration (Tahap Desain)

Pada tahap ini dilakukan perancangan perangkat lunak mulai dari menspesifikasikan fitur perangkat lunak hingga perilisan prototipe dari perangkat lunak..

c. Tahap Contruction (Tahap Implementasi dan Testing)

Implementasi perangakat lunak yang telah dibuat dilakukan pada tahap ini.

d. Tahap Transition (Tahap Deployment)

Instalasi, deployment dan sosialisasi perangkat lunak dilakukan pada tahap ini.

IMPLEMENTASI ALGORITMA DAN PROGRAM

1. Lingkungan Implementasi

Pada pembahasan lingkungan implementasi meliputi pembahasan spesifikasi hardware yang digunakan, perangkat lunak, perangkat pembangun dan tools yang digunakan untuk membuat sistem Aplikasi Enkripsi.

2. Hardware

Sistem aplikasi enkripsi dibuat dengan perangkat keras yang memiliki spesifikasi sebagai berikut :

Processor : Intel(R) Celeron(R) CPU 1007U @1,50 Ghz
Memory : 2048 MB RAM
Adapter Video : Intel(R) HD Graphics
Chipset Type : Intel(R) HD Graphics Family
Display Mode : 1366 x 768 (32 bit) (60Hz)
Monitor : 14” (Inch)

3. Software

Sistem aplikasi Enkripsi dibuat dengan perangkat lunak, perangkat pembangun dan aplikasi pembantu sebagai berikut:

Sistem Operasi : Microsoft Windows 8.1

Perangkat pembangun : XAMPP, MySQL

Aplikasi pembantu : Notepad ++, web browser google chrome

HASIL DAN PEMBAHASAN

Tampilan Aplikasi

Pada saat aplikasi dijalankan maka akan muncul tampilan layar utama.

Gambar 7 : Halaman Utama

Gambar 7 menunjukkan tampilan halaman utama aplikasi dengan pilihan dua menu utama Home dan Login

Gambar 8 : Halaman Login

Gambar diatas menampilkan layar login aplikasi, Jika dalam melakukan input username dan password tidak diisi atau salah, maka aplikasi akan memberikan peringatan " password yang anda masukksan salah ", jika input benar maka aplikasi akan memberikan peringatan " password yang anda masukkan benar.

Gambar 9 : Halaman Menu

Setelah berhasil melalui proses login selanjutnya akan berpindah pada layar halaman menu seperti Gambar 9, pada layar terdapat tampilan menu aplikasi untuk proses enkripsi dan dekripsi data.

Gambat 10 : Layar Proses Enkripsi

Untuk memulai proses enkripsi pada tampilan layar gambar 10, user terlebih dahulu memilih file yang akan di proses, dengan cara klik tombol pilih file, kemudian pilih file dengan tipe docx, xlsx, atau pdf. Setelah file dipilih, input kata kunci pada kolom yang disediakan dengan digit minimun adalah enam digit yang merupakan gabungan angka dan huruf atau hanya dengan masing-masing jenis, kata kunci tersebut yang akan menjadi master key dalam melakukan proses dekripsi. Setelah proses dilakukan kemudian klik tombol enkrip,maka aplikasi mulai menjalankan proses enkripsi. Setelah proses enkripsi selesai, tindakan selanjutnya adalah mengklik tombol download, sehingga data yang sudah di enkripsi dapat di ambil dalam folder download sistem komputer yang digunakan. Data hasil dari setiap enkripsi akan diberi kode nama awalan dengan enkrip yang diikuti empat digit angka. Setiap hasil enkripsi pasti memiliki kode angka yang berbeda walaupun melakukan proses enkripsi berulang-ulang pada data yang sama. Tujuan pemberian kode tersebut untuk membedakan waktu proses yang dilakukan, sehingga dapat diingat dengan mudah.

Gambar 11 : Layar Proses Dekripsi

Untuk melakukan proses dekripsi, user terlebih dahulu memilih file yang sudah di enkripsi pada proses enkripsi. Klik tombol pilih file, kemudian cari data yang sudah di enkrip dengan keterangan "Enkrip_(4 digit angka)_namafile". Selanjutnya masukkan kata kunci, kata kunci yang digunakan untuk dekripsi harus sama dengan kata kunci yang digunakan untuk proses enkripsi. Setelah tata cara dekripsi dilaksanakan, klik tombol dekrip pada layar kemudian aplikasi akan melakukan proses dekripsi data. Setelah proses selesai, tindakan yang sama dilakukan seperti pada proses enkripsi yaitu klik tombol download dan data dapat di ambil pada folder download sistem komputer yang digunakan

PENGUJIAN APLIKASI

Dalam penelitian ini dilakukan pengujian aplikasi antara lain

a. Membandingkan data file sebelum di proses dan sesudah diproses pada aplikasi.
b. Mencoba membuka data yang sudah dienkripsi, untuk melihat hasil dari proses enkripsi lalu membandingkannya dengan data asli yang belum di enkripsi.
c. Mencoba memproses semua jenis data untuk membuktikan batasan dari aplikasi yang dibuat.

NO	Nama Data	Jenis Data	Ukuran Data	Status
1	28D4D5AD	xlxs	10.6 KB(10.916 bytes)	Berhasil
2	Avril Lavigne	JPEG	134 KB(137.386 bytes)	Gagal
3	BAB I	docx	29.0 KB(29.730 bytes)	Berhasil
4	PLOT	text	1.8 KB(1.851 bytes)	Gagal
5	Fight Song	MPEG	1.60 MB(1.687.760 bytes)	Gagal
6	SKRIPSI 20	pdf	771 KB(790.330 bytes)	Berhasil

Gambar 12 : Pengujian Data

ANALISA HASIL PENGUJIAN

Analisa hasil dari pengujian yang dilakukan pada aplikasi enkripsi diantaranya sebagai berikut :

1. Data file yang telah di proses enkripsi mengalami peningkatan pada ukuran file sehingga menjadi lebih besar, hal tersebut disebabkan oleh proses enkripsi dalam melakukan perlindungan keamanan data dengan memblokir semua akses atau ujicoba membuka data file serta karena adanya penyisipan kata kunci yang digunakan untuk melakukan proses dekrip data yang sudah di proses.
2. Tipe data yang dapat diproses oleh aplikasi yang telah dibuat hanya berjumlah tiga jenis yaitu : docx, xlxs, dan pdf. Selain jenis tersebut maka data tidak dapat di proses oleh aplikasi enkripsi. Jenis data yang dipilih berdasarkan kebutuhan user dan juga untuk mempermudah dalam tujuan pembuatan aplikasi serta penggunaannya.
3. Jenis tipe data yang dapat diproses dapat melebihi tiga jenis data, hal tersebut dapat dilakukan dengan menyesuaikan kebutuhan dari user yang menggunakan aplikasi

tersebut. Jika aplikasi dibuat untuk dapat memproses semua jenis type data, akan berdampak pada kinerja aplikasi tersebut. Dimana sebagian jenis data tidak terlalu penting untuk di proses enkripsi.

KESIMPULAN

Berikut kesimpulan mengenai Implementasi Kriptografi Algoritma AES Serta Algoritma Kompresi Huffman dengan Pemrograman PHP.

1. Input dan output dari algoritma *AES* terdiri dari urutan data sebesar 128 bit. Urutan data yang sudah terbentuk dalam satu kelompok 128 bit tersebut disebut juga sebagai blok data atau *plaintext* yang nantinya akan dienkripsi menjadi chiphertext. Chiper *key* dari *AES* terdiri dari *key* dengan panjang 128 bit, 192 bit, atau 256 bit. Perbedaan panjang kunci akan mempengaruhi jumlah round yang akan iimplementasikan pada algoritma *AES.* Proses enkripsi pada algoritma AES terdiri dari 4 jenis transformasi *byte*s, yaitu *SubBytes*, *ShiftRows*, *Mixcolumns*, dan *AddRoundKey*. Pada awal proses enkripsi, input yang telah dikopikan ke dalam state akan mengalami transformasi *byte AddRoundKey*.
2. Algoritma AES dapat diterapkan dengan membangun sebuah sistem olah data sederhana berbasis web yang dapat berjalan pada web browser umum digunakan dengan bantuan tools tambahan berupa aplikasi Xampp dan MySql.
3. Algoritma Aes dapat digunakan untuk semua jenis data komputerisasi, pengembangan dan implementasi disesuaikan dengan tingkat kebutuhan masing – masing user client.

SARAN

Setelah melakukan analisa dan memberikan suatu kesimpulan, maka penulis akan memberikan saran yang dapat dijadikan bahan masukan yang dalam membangun sistem yang lebih baik di masa yang akan datang, yaitu sebagai berikut :

1. Sistem yang penulis kembangkan hanya memproses jenis data yang disesuaikan dengan kebutuhan user, jika kedepan terdapat mahasiswa yang melakukan penelitian dengan tema yang sama, sistem dapat dikembangkan dengan kemampuan yang lebih baik dengan cakupan data yang lebih banyak dan pengguna yang lebih multi user.
2. Sistem yang dibuat berjalan pada browser dengan sistem localhost. Penulis berharap kedepan terdapat mahasiswa yang dapat mengembangkan sistem tersebut untuk berjalan pada basis web dengan koneksi internet sehingga penggunaan multiuser dapat tercapai.

DAFTAR PUSTAKA

[1] Ariyus Dony, 2006, ”Pengantar Ilmu Kriptografi” Penerbit Informatika, Bandung

[2] r. Kurniawan Yusuf, MT., 2004, “Kriptografi: Keamanan Internet dan Jaringan Komunikasi” Penerbit Informatika Bandung, Yogyakarta

[3] Kristanto Andi, 2004, ”Memahami Model Enkripsi dan Keamanan Data” Kerjasama Wahana Komputer Semarang dengan Andi Offset Yokyakarta

[4] Munir Rinaldi, 2008, ”Belajar Ilmu Kriptografi” Penerbit Andi, Yogyakarta

[5] IEEE, 2011, IEEE P1619/D20 Draft Standard for Cryptographic Protection of Data on Block-Oriented Storage Devices, New York, USA

[6] Stallings, William 2011, *Cryptography and Network Security : Principles and Practice*, Prentice Hall, United State of Amerika