PENGAMANAN DATA MENGGUNAKAN METODA ENKRIPSI EINSTEIN

Semuil Tjiharjadi[1], Marvin Chandra Wijaya[2]
Jurusan Sistem Komputer, Fakultas Teknik, Universitas Kristen Maranatha
Jl. Suria Sumantri 65, Bandung 40164
Telp. (022) 2012186 ext. 228, Faks. (022) 2012186 ext. 230
E-mail: semuiltj@maranatha.edu ; marvinchw@gmail.com

Abstrak
Dalam proses demokrasi maka semua orang bebas untuk berbicara mengutarakan pendapat dan pandangannya berdasarkan pribadi dan perasaannya, tentu saja dengan melihat bahwa hak demokrasinya tersebut tidak melanggar hak demokrasi orang lain. Kebebasan berkomunikasi ini juga termasuk kebebasan untuk berbicara dengan orang yang diinginkan. Untuk itu akan sangat mengganggu bila isi pembicaraan terutama yang menggunakan teknologi informasi ternyata bocor kepada orang yang tidak berhak dan secara demokrasi ini melanggar haknya, padahal dalam teknologi informasi yang berkembang secara sangat pesat ini ternyata tidak dibarengi dengan penggunaan alat untuk pengamanan data yang tepat dalam sistem informasi.
Salah satu teknik pengamanan data informasi di dunia internet adalah penggunaan teknik algoritma kriptografi. Suatu algoritma kriptografi berisi fungsi-fungsi matematika yang digunakan untuk melakukan proses enkripsi dan dekripsi. Algoritma kriptografi yang digunakan merupakan jenis algoritma kriptografi simetrik yang menggunakan kunci rahasia yang sama untuk proses enkripsi dan dekripsinya. Pada makalah ini dipaparkan penggunaan algoritma kriptografi Einstein sebagai salah satu cara untuk mengamankan data. Pada algoritma Einstein, terdapat proses acak (random) yang menggunakan metoda kongruensial linear. Algoritma Einstein mempunyai kelebihan dalam melakukan proses enkripsi dan dekripsi pada hampir semua jenis file yang umum digunakan. Algoritma Einstein bisa diimplementasikan untuk semua ukuran file.

***Kata kunci**: Einstein, enkripsi*

1. PENDAHULUAN

Keamanan pada dunia internet menjadi suatu kebutuhan dan keharusan yang sangat penting dalam semua aspek kehidupan bermasyarakat. Keamanan data informasi merupakan faktor utama dan terdepan yang menentukan apakah data informasi tersebut masih berguna dan dapat digunakan. Tingkat keamanan data informasi yang akan digunakan bermacam – macam bergantung pada kegunaan data informasi tersebut. Pada dunia *e-commerce*, data informasi yang digunakan dan dipertukarkan mempunyai kriteria tingkat keamanan yang tinggi agar tidak terjadi penyalahgunaan dan pembajakan.

Salah satu teknik pengamanan data informasi di dunia internet adalah teknik kriptografi Einstein. Suatu Algoritma kriptografi berisi fungsi-fungsi matematika yang digunakan untuk melakukan proses enkripsi dan dekripsi. Dasar matematis yang mendasari proses enkripsi dan dekripsi adalah relasi antara dua himpunan yaitu yang berisi elemen *plaintext* dan yang berisi elemen *ciphertext*.

2. TINJAUAN PUSTAKA

Kriptologi (*Cryptology*) adalah salah satu ilmu pengetahuan yang mempelajari suatu cara komunikasi yang aman, yang meliputi kriptografi (*Cryptography*) dan kriptoanalisis (*Cryptanalysis*). Kriptografi adalah cabang dari kriptologi yang mengenai desain algoritma untuk proses enkripsi dan dekripsi, yang dimaksudkan untuk memastikan kerahasiaan dan atau keautentikan dari suatu pesan. Kriptografi tidak berarti hanya memberikan keamanan informasi saja, namun lebih ke arah teknik-tekniknya atau secara garis besar kriptografi dapat diartikan sebagai suatu seni dan ilmu untuk menjaga keamanan dari suatu pesan. Kriptoanalisis adalah cabang dari kriptologi yang membahas tentang cara menguraikan pesan terenkripsi (*Ciphertext*) untuk mendapatkan informasi, atau menempa ulang informasi terenkripsi sehingga informasi itu dianggap autentik.

Ada tiga tujuan mendasar dari ilmu kriptografi ini yaitu :

1. Kerahasiaan (*Confidentiality*) adalah ketentuan yang digunakan untuk mengamankan isi informasi dari siapapun kecuali yang memiliki otoritas untuk memilikinya.
2. Integritas Data (*Data Integrity*) adalah ketentuan yang berhubungan dengan pengamanan dari perubahan data secara tidak sah. Untuk menjaga integritas data, sistem harus memiliki kemampuan untuk mendeteksi suatu keadaan percobaan manipulasi data oleh pihak-pihak yang tidak berhak, antara lain penyisipan, penghapusan, dan pensubsitusian data lain ke dalam data yang sebenarnya.
3. Autentikasi (*Authentication*) adalah ketentuan yang berhubungan dengan identifikasi dimana informasi yang dikirimkan harus diautentikasi keasliannya.

Sistem kriptografi terbagi menjadi 3 dimensi yang berbeda, antara lain :

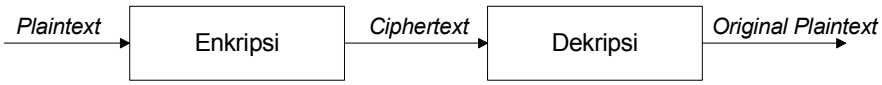
a. Tipe metoda yang digunakan dalam transformasi *plaintext* menjadi *ciphertext*. Semua algoritma enkripsi berdasarkan pada 2 prinsip dasar, yaitu subtitusi, dimana setiap elemen dalam *plaintext* dipetakan dengan elemen yang lain, dan tranposisi, dimana elemen dalam *plaintext* disusun ulang. Kebutuhan fundamentalnya adalah tidak adanya informasi yang hilang. Kebanyakan sistem yang ada melibatkan tingkatan yang berjenjang dalam subtitusi dan transposisi.
b. Kunci yang digunakan. Jika pengirim dan penerima menggunakan kunci yang sama, sistem ini disebut simetrik, kunci tunggal, kunci rahasia, atau enkripsi konvensional. Jika pengirim dan penerima menggunakan kunci yang berbeda, sistem ini disebut asimetrik, kunci ganda, atau enkripsi kunci publik.
c. Metoda pemrosesan *plaintext*. *Cipher block* memproses masukan satu blok dalam satu waktu, menghasilkan keluaran satu blok dari input satu blok. *Stream cipher* memroses elemen masukan secara terus menerus, menghasilkan keluaran satu elemen pada satu waktu.

Dalam kriptografi suatu pesan yang akan dirahasiakan akan disandikan dengan menggunakan suatu algoritma. Pesan yang telah disandikan disebut *plaintext* dan pesan yang sudah diacak atau disandikan disebut *ciphertext*. Proses untuk mengkonversi *plaintext* menjadi ciphertext disebut enkripsi (*encrypt*) dan proses untuk mengembalikan *plaintext* dari *ciphertext* disebut dekripsi (*decrypt*).

Enkripsi dan dekripsi merupakan fungsi transformasi antara himpunan-himpunan tersebut. Apabila elemen-elemen *plaintext* dinotasikan dengan M, elemen-elemen *ciphertext* dinotasikan dengan C, sedang untuk proses enkripsi dinotasikan dengan E, dekripsi dengan notasi D. Notasi matematika dari proses ini adalah:

- Enkripsi : E(M) = C
- Dekripsi : D(C) = D(E(M)) = M

Dengan diagram blok, proses enkripsi dan dekripsi dapat digambarkan sebagai berikut :



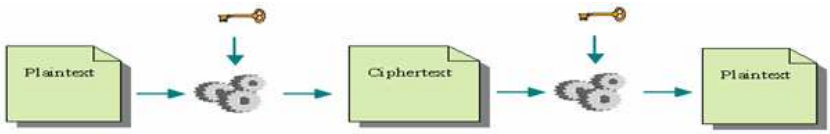
Gambar 1. Diagram Blok Proses Enkripsi dan Dekripsi

Pada operasi enkripsi dan dekripsi dibutuhkan suatu kunci yang gunanya untuk menjaga kerahasiaan cara kerja dari algoritma enkripsi dan dekripsi. Algoritma enkripsi yang didasarkan pada kunci digolongkan menjadi dua bagian :

1. Algoritma Simetrik (*Symmetric Algorithms*), dimana kunci yang dipakai untuk proses enkripsi maupun proses dekripsi sama. Algoritma ini dapat disebut juga *secret-key algorithms* atau *one-key algorithms.*
2. Algoritma Asimetrik (*Asymmetric Algorithms*), dimana menggunakan kunci yang berbeda yaitu kunci publik (*public key*) untuk melakukan proses enkripsi dan kunci pribadi (*private key*) untuk melakukan proses dekripsi.

Algoritma simetrik pada proses enkripsi dan dekripsi dapat disimbolkan secara matematis dan digambarkan sebagai berikut :

- Enkripsi : $E_K(M) = C$
- Dekripsi : $D_K(C) = D_K (E_K (C)) = M$



Gambar 2. Algoritma Simetrik

Algoritma asimetrik (*asymmetric algorithm*) menggunakan kunci enkripsi dan kunci dekripsi yang berbeda. Kunci enkripsi dapat disebarkan kepada umum dan dinamakan sebagai kunci publik (*public key*), sedangkan kunci dekripsi disimpan dan dirahasiakan untuk digunakan sendiri dan dinamakan sebagai kunci pribadi (*private key*). Faktor kerahasiaan pada algoritma ini sangat tergantung pada kerahasiaan kunci pribadi (*private key*) yang digunakan. Faktor validitas bergantung pada keamanan kunci pribadi (*private key*) tersebut. Oleh karena itulah, algoritma ini dikenal pula dengan nama algoritma kunci publik (*public key algorithm*).

Algoritma asimetrik ini menggunakan ukuran kunci, baik kunci publik maupun kunci pribadi, dengan ukuran yang lebih panjang atau nilai yang lebih besar jika dibandingkan dengan kunci rahasia pada algoritma simetrik. Algoritma asimetrik lebih lamban dalam hal penggunaan waktu dan hanya berguna secara efektif pada pemrosesan data dengan jumlah yang sedikit.

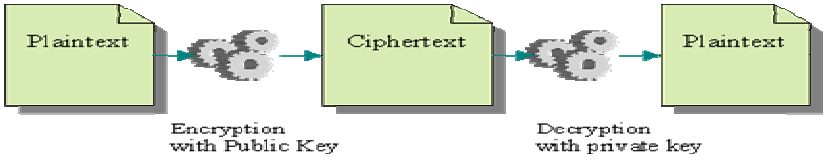
Algoritma asimetrik atau kriptografi kunci publik (*public key cryptography*) terbagi menjadi dua cabang utama, yaitu

- ❖ Enkripsi kunci publik (*public key encryption*), yaitu dimana pesan yang terenkripsi oleh pengguna kunci publik tidak bisa didekripsi oleh orang lain kecuali pengguna kunci pribadi lain yang telah diserahkan oleh pengguna kunci publik.
- ❖ Tanda tangan digital (*digital signature*), yaitu dimana pesan yang telah ditandai dengan kunci pribadi pengguna bisa diperiksa kebenarannya oleh orang lain yang bisa menggunakan kunci publik pengguna untuk bisa membuktikan bahwa pengguna telah menandainya dan pesan tidak mengalami perubahan.

Algoritma asimetrik pada proses enkripsi dan dekripsi dapat disimbolkan secara matematis dan digambarkan sebagai berikut :

- ∝ Enkripsi : $E_{K1}(M) = C$
- ∝ Dekripsi : $D_{K2}(C) = D_{K2}(E_{K1}(C)) = M$

Kunci publik dinotasikan dengan *K1* sedangkan Kunci privat dinotasikan dengan *K2*.



Gambar 3. Algoritma Asimetrik

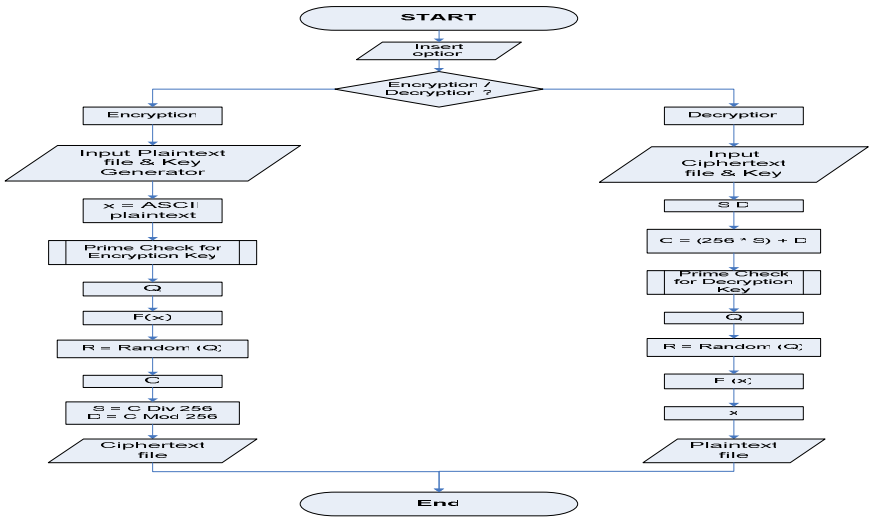
3. METODA PENELITIAN

Algoritma Enstein menggunakan teknik perkalian antara dua buah faktor prima dari sebuah bilangan kunci, dan teknik penggantian bit (*bit substitution*) dengan fungsi XOR. Cipher text akan disimpan dalam format 24 bit data, dengan 16 bit pertama sebagai *sign*, dan 8 bit kedua sebagai data.

Perangkat lunak pengamanan data dengan algoritma kriptografi Einstein direalisasikan menggunakan tipe bahasa pemograman Visual Basic yang terdapat pada perangkat lunak Microsoft Visual Basic.

Pada saat perangkat lunak pengamanan data algoritma kriptografi Einstein dimulai, akan terlihat bahwa perangkat lunak ini terbagi menjadi 2 bagian utama, yaitu

1. Enkripsi
   Bagian enkripsi ini terdiri dari mendapatkan masukan *plaintext file* dan masukan pembangkit kunci, pemeriksaan bilangan prima kunci, perhitungan dari *plaintext file* menjadi *plaintext*, perhitungan dari *plaintext* menjadi hasil enkripsi dasar, perhitungan hasil enkripsi dasar menjadi *ciphertext*, dan menampilkan *ciphertext file*.
2. Dekripsi
   Bagian dekripsi ini terdiri dari mendapatkan masukan *ciphertext file* dan masukan kunci, pemeriksaan bilangan prima kunci, perhitungan dari *ciphertext file* menjadi *ciphertext* yang sebenarnya, perhitungan dari *ciphertext* menjadi hasil enkripsi dasar, perhitungan dari hasil enkripsi dasar menjadi plaintext yang sebenarnya, dan perhitungan *plaintext* menjadi *plaintext file*, dan menampilkan *plaintext file*.



Gambar 4. Diagram alir perangkat lunak algoritma Einstein

Algoritma kriptografi Einstein terdiri dari lima (5) bagian utama, antara lain :

1. Program enkripsi
2. Program pemeriksaan kunci untuk enkripsi
3. Program dekripsi
4. Program pemeriksaan kunci untuk dekripsi
5. Program acak (*random*)

Bagian program enkripsi terdiri dari mendapatkan masukan *plaintext file* dan masukan pembangkit kunci, pemeriksaan bilangan prima kunci, perhitungan dari *plaintext file* menjadi *plaintext*, perhitungan dari *plaintext* menjadi hasil enkripsi dasar, perhitungan hasil enkripsi dasar menjadi *ciphertext*, dan menampilkan *ciphertext file*.

Masukan *plaintext file* diambil nilai ASCII-nya untuk menjadi *plaintext* (x). Masukan pembangkit kunci yang telah melalui pemeriksaan bilangan prima dan mengandung faktor bilangan prima (A,B) akan digunakan menjadi kunci rahasia (Q).

Algoritma Enstein menggunakan rumus dasar :

$$Q = A * B$$

dengan :
Q : bilangan kunci untuk proses enkripsi.
A dan B : faktor prima dari Q, dan A > B.

Untuk mendapatkan hasil enkripsi dasar (F(x)) digunakan rumus :

$$F(x) = \left(Q * \frac{A}{B} \right) + \left(Q * x * \left(A + B \right) \right)$$

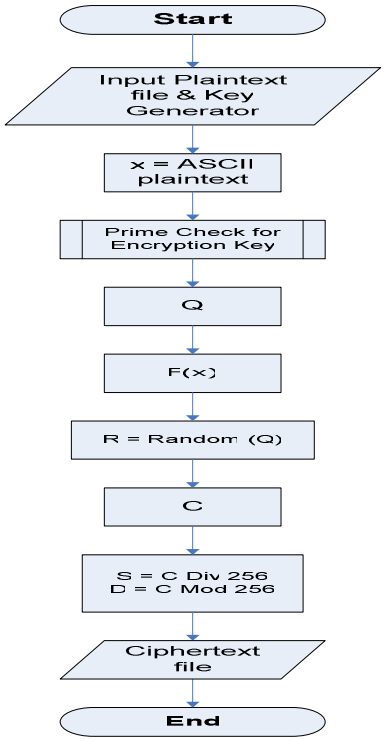
dengan :
Q : bilangan kunci untuk proses enkripsi.
A dan B : faktor prima dari Q, dan A > B.
F(x) : hasil dari enkripsi dasar dari plain text.
X : nilai ASCII dari plain text.

Proses berikutnya adalah proses random kunci rahasia (Q) yang digunakan untuk mendapatkan *ciphertext* (C) dengan rumus :

$$C = F(x) \text{ XOR Random}(Q)$$

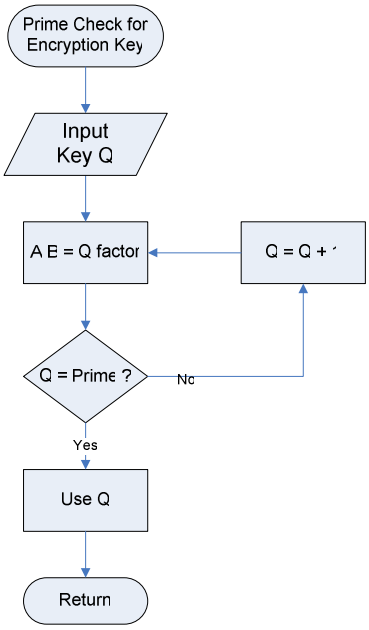
Untuk menampilkan *ciphertext* (C) menjadi *ciphertext file* (S,D) melalui rumus operasi matematis :

$$S = C \textit{ Div } 256$$
$$D = C \textit{ Mod } 256$$



Gambar 5. Diagram alir proses enkripsi

Masukan kunci rahasia harus mengandung faktor bilangan prima (A,B). Masukan kunci rahasia yang mengandung faktor bilangan prima (A,B) akan menjadi kunci rahasia (Q) yang digunakan pada proses enkripsi dan dekripsi program algoritma kriptografi Einstein.

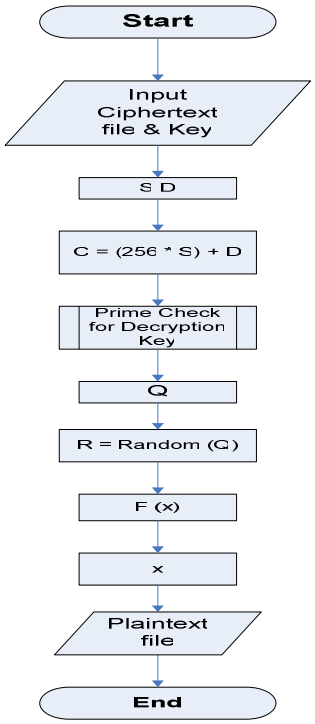


Gambar 6. Diagram alir pemeriksaan kunci untuk enkripsi

Bagian dekripsi ini terdiri dari mendapatkan masukan *ciphertext file* dan masukan kunci, pemeriksaan bilangan prima kunci, perhitungan dari *ciphertext file* menjadi *ciphertext* yang sebenarnya, perhitungan dari *ciphertext* menjadi hasil enkripsi dasar, perhitungan dari hasil enkripsi dasar menjadi plaintext yang sebenarnya, dan perhitungan *plaintext* menjadi *plaintext file*, dan menampilkan *plaintext file*.

Untuk dapat membaca *ciphertext file* (S,D), harus diubah terlebih dahulu menjadi ciphertext (C) dengan rumus :

$$C = (S * 256) + D$$



Gambar 7. Diagram alir program dekripsi

Masukan kunci rahasia (Q) diperiksa terlebih dahulu dan kemudian didapatkan faktor bilangan primanya (A,B). Masukan kunci rahasia (Q) yang telah diperiksa melalui proses *random* yang kemudian akan digunakan untuk mendapatkan hasil enkripsi dasar (F(x)) dengan rumus :

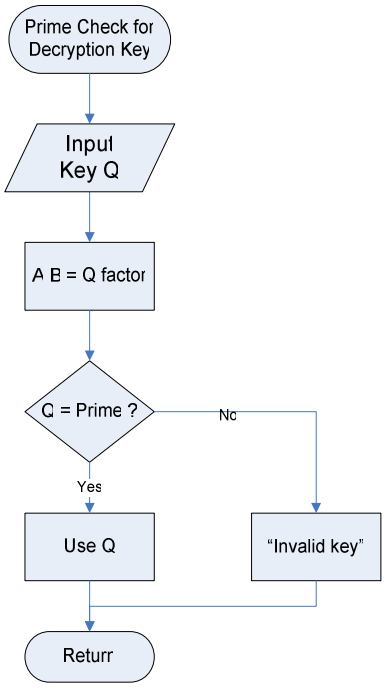
F(x) = C XOR Random(Q).

Untuk mendapatkan *plaintext* (x), digunakan rumus :

$$x = \frac{\left(F(x) - \left(Q * \frac{A}{B}\right)\right)}{(Q * (A + B))}$$

Nilai *plaintext* (x) merupakan nilai ASCII dari *plaintext file*.

Masukan kunci (Q) harus melalui pemeriksaan untuk mengetahui faktor bilangan primanya (A,B). Untuk keamanan, masukan kunci rahasia yang tidak sesuai akan mengakibatkan program algoritma kriptografi Einstein dimatikan.



Gambar 9 Diagram alir pemeriksaan kunci untuk dekripsi

Dalam algoritma Einstein, kunci enkripsi digunakan sebagai benih (*seed*) untuk mendapatkan bilangan acak. Proses untuk mendapatkan bilangan acak ini menggunakan rumus :

R = ((V * 214013) + 2531011) Mod 1023

dengan :

R : hasil dari bilangan acak.

V : *seed*, benih untuk mendapatkan bilangan acak.

Dalam proses enkripsi ini, hasil bilangan acak (R) merupakan benih (V) untuk bilangan acak selanjutnya, dengan mengeliminsai nilai benih 0 (V = 0), karena untuk nilai benih 0 (V = 0), maka hasil dari bilangan acak akan sama dengan 0. Jika terjadi V = 0, maka nilai V diganti dengan angka 13.

4. HASIL DAN PEMBAHASAN

Perangkat lunak tambahan yang digunakan untuk memeriksa hasil proses enkripsi dan dekripsi pada algoritma Einstein ini adalah EditPlus Text Editor versi 2.11 dan Nero Wave Editor versi 6 agar hasil analisa dan pembuktian lebih akurat.

Hasil pengamatan pengujian perangkat lunak ini terbagi dalam beberapa kondisi *plaintext*, nilai pembangkit kunci yang digunakan, ukuran file sebelum dan sesudah proses enkripsi dekripsi, serta waktu yang dibutuhkan pada proses enkripsi dekripsi.

Untuk analisa, dilakukan pengamatan terhadap berbagai jenis tipe *file* yang diambil masing-masing dua (2) *file* dari tiap *file* yang diuji dan diamati.

Tabel 1. Hasil pengamatan

Plaintext		Ciphertext (encrypted plaintext)		Plaintext (decrypted ciphertext)	
Filename	**Size (bytes)**	**Size (bytes)**	**Time duration (*hh*:*mm*:*ss*)**	**Size (bytes)**	**Time duration (*hh*:*mm*:*ss*)**
Gbr.jpg	4474	53798	00:00:14	4474	00:00:14
0lastjud.jpg	198387	2380761	00:10:20	198387	00:10:20
Paradise-sade.mp3	3731121	44773575	03:14:20	3731121	03:14:24
King of convenience.mp3	1293607	15523413	01:07:22	1293607	01:07:23
Kitchen.swf	457048	5484697	00:23:50	457048	00:23:48
Corridor.swf	673589	8083190	00:35:05	673589	00:35:05
Movie-setan.zip	2890481	34685865	02:30:33	2890481	02:30:32
savetheantilope.mpg	4822432	57869309	04:11:10	4822432	04:11:12
Roberto_carlos.mpg	2684372	32212588	02:19:49	2684372	02:19:48
Scaredmirrormonkey.wmv	218922	2627191	00:11:24	218922	00:11:24
AUDREY.pps	286720	3440776	00:14:56	286720	00:14:56
A_SmallTruthToMakeLife.pps	168448	2021525	00:08:46	168448	00:08:46
counter_terrorism_strategy.pdf	274677	3296268	00:14:18	274677	00:14:18
paper_w.pdf	758795	9105665	00:39:31	758795	00:39:32

Pengujian perangkat lunak ini dilakukan dengan sistem operasi Windows XP Professional pada CPU dengan spesifikasi sebagai berikut :

- Prosesor AMD Athlon XP 1600+ (1,04 Ghz)
- *Motherboard* ECS K7S6A
- Kartu *memory* Visipro DDR-SDRAM 256MB PC2100
- Kartu grafis MSI GeForce3 TI 128MB
- *Harddisk* Maxtor 60MB

Pada pengamatan hasil percobaan yang dilakukan pada berbagai macam tipe *file – file* yang umum digunakan, terlihat bahwa *plaintext file* dan hasil *ciphertext file* yang dilakukan proses dekripsi mempunyai ukuran file yang sama besar. Sedangkan ukuran *ciphertext file* yang dihasilkan dari proses enkripsi pada masukan *plaintext file* mempunyai ukuran *file* yang lebih besar daripada *plaintext file* tersebut dengan tingkat persentase rata – rata lebih dari 1200%.

Besaran nilai masukan kunci yang digunakan tidak mempengaruhi ukuran *ciphertext file*, baik puluhan, ratusan, maupun ribuan. Ukuran masukan kunci yang besar, dimana nilainya diatas 10000, akan menyebabkan ketidakstabilan proses sehingga program tidak berjalan. Karena itu, masukan kunci pada program kriptografi algoritma Einstein mempunyai keterbatasan yang tidak bisa melebihi nilai 9999.

Pada proses enkripsi dan dekripsi, didapatkan durasi waktu yang tidak jauh berbeda.

5. KESIMPULAN DAN SARAN

1. Perancangan perangkat lunak untuk mengamankan data dengan algoritma kriptografi Einstein telah berhasil direalisasikan untuk tipe-tipe *file* yang umum digunakan, seperti *.jpg, *.mp3, *.swf, *.zip, *.mpg, *.wmv, dan sebagainya.
2. Lama waktu yang dibutuhkan dalam proses dekripsi hampir sama dengan proses enkripsinya.
3. Berdasarkan riset yang pernah dilakukan terhadap metoda enkripsi yang lain maka metoda ini memiliki kecepatan yang cukup baik sehingga dapat dikombinasikan dengan sistem metoda enkripsi yang memiliki kecepatan lebih rendah namun memiliki keamanan lebih tinggi.

6. DAFTAR PUSTAKA

[1] Davies, 1989, D. dan W. Price, *Security for computer networks*, Wiley, New Yorks.
[2] Stallings, William, 2003, *Cryptography and network security : Principles and practices*, Prentice Hall, New Jersey.
[3] Tsudik, G. , 1992, *Message authentication with one-way hash functions*, INFOCOM 92.
[4] http://www.answers.com
[5] http://en.wikipedia.org
[6] ms-help://MS.MSDNQTR.2004JAN.1033