PEMODELAN GERAKAN CHESS KNIGHT DALAM MASALAH SISTEM PRODUKSI

Ari Usman[1], Yuyun Dwi Lestari[2]
[1, 2, 3]Program Studi Teknik Informatika, Fakultas Teknik dan Komputer, Universitas Harapan Medan

e-mail: [1]ariusman@unhar.ac.id, [2]yuyundwilestari@unhar.ac.id

ABSTRAK

Kecerdasan Buatan merupakan salah satu bagian ilmu komputer yang mempelajari tentang bagaimana cara membuat agar komputer dapat melakukan pekerjaan seperti yang dilakukan oleh manusia. Biji kuda dalam papan catur memiliki pergerakan menyerupai huruf L. Biji catur ini merupakan salah satu biji yang sangat sulit digerakkan dan sering juga merupakan biji yang paling berbahaya apabila tidak diperhatikan secara seksama setiap pergerakannya. Simulasi dari permasalahan ini menyediakan sebuah papan catur berukuran n x n. Sasaran (*goal*) dari permasalahan ini adalah menggerakkan sebuah biji kuda dari suatu posisi tertentu pada papan catur ke posisi tujuan yang diinginkan dengan mensimulasikan semua solusi pergerakan terpendek yang mungkin untuk menuju ke posisi tujuan tersebut. Permasalahan ini juga merupakan salah satu masalah klasik dalam *artificial intelligence* (AI). Penyelesaian permasalahan ini dapat menggunakan bantuan sistem produksi dan pohon pelacakan.
Kata kunci: Biji Catur, Kecerdasan Buatan, Goal

ABSTRACT

Artificial Intelligence is a part of computer science that studies how to make computers able to do work like humans do. The horse checker on a chess board has a movement that resembles the letter L. This chess checker is one of the pieces that is very difficult to move and is often the most dangerous checker if you don't pay close attention to each movement. The simulation of this problem provides an n x n chessboard. The goal of this problem is to move a horse from a certain position on the chess board to the desired goal position by simulating all the shortest possible movement solutions to get to that goal position. This problem is also one of the classic problems in artificial intelligence (AI). Solving this problem can use the help of production systems and tracking trees.
***Keywords**: Chess Checkers, Artificial Intelligence, Goal*

1. PENDAHULUAN

Perkembangan ilmu dan teknologi komputer telah mempengaruhi segala aspek kehidupan manusia seperti di bidang pendidikan. Informasi dan data dapat dengan mudah dan cepat untuk dikirim ke konsumen melalui jaringan komputer. Terdapat berbagai bentuk pesan rahasia seperti pesan teks (dalam bentuk *file*), pesan citra, pesan audio dan pesan video

yang tentu saja menimbulkan risiko jika informasi dan data yang dikirim bisa diakses oleh pihak yang tidak berhak sehingga mengakibatkan kebocoran data.
Dalam penelitian ini penulis tertarik untuk mengamankan file teks, dalam mengamankan sebuah file teks terdapat berbagai macam cara, salah satunya menggunakan teknik kriptografi. Kriptografi merupakan ilmu yang mempelajari teknik-teknik matematika yang berhubungan dengan aspek keamanan informasi, seperti kerahasiaan data, keabsahan data, integritas data, serta autentikasi data. Dengan menggunakan teknik kriptografi maka data akan lebih terjaga keamanannya hingga data tersebut akan sampai ke tujuannya dengan tepat. Ada banyak metode yang digunakan dalam kriptografi, salah satunya adalah algoritma *Merkle-Hellman Knapsack*. *Merkle-Hellman Knapsack* merupakan kriptosistem yang menggunakan algoritma *asymmetries* [3]. Panjang kunci yang digunakan antara 8 sampai 72 bit. Dalam kriptosistem *Merkle-Hellman Knapsack*, 'penyamaran' diakhiri dengan penelusuran diberikan $s_1, s_2, \ldots s_n$ merupakan sebuah himpunan *super-increasing*. Memilih suatu bilangan prima p yang lebih besar dari pada jumlah semua si, dan suatu bilangan b dengan 1 < b < p.
Berdasarkan penelitian [4] "*Cryptography Asymmetries Merkle-Hellman Knapsack* Digunakan untuk Enkripsi dan Dekripsi Teks" disimpulkan bahwa metode *Merkle-Hellman Knapsack* memiliki kelebihan dalam proses pendistribusian kunci pada media yang tidak aman seperti internet dan tidak memerlukan kerahasiaan. Karena kunci yang didistribusikan adalah kunci publik. Sehingga jika kunci ini sampai hilang atau diketahui oleh orang lain yang tidak berhak, maka pesan sandi yang dikirim akan tetap aman, karena kunci *private* tidak didistribusikan. Penelitian lainnya yang dibuat oleh [5] "Keamanan Data dengan Metode Kriptografi Kunci Publik" disimpulkan metode ini mampu menggabungkan algoritma kunci publik dengan algoritma simetrik untuk memperoleh keunggulan-keunggulan pada masing-masing algoritma. Selanjutnya yaitu penelitian oleh [15]"Kriptografi Gabungan Menggunakan Algoritma *Mono Alphabetic* dan *One Time Pad*" disimpulkan bahwa keamanan algoritma pengenkripsian ini sangat bergantung pada kerahasiaan kunci rahasia (*secret key*) dan *pad* yang digunakan baik dalam mengenkripsi maupun mendekripsi data dan informasi. Berdasarkan penjabaran diatas maka penulis bermaksud untuk melakukan penelitian dengan membuat batasan masalah dan tujuan penelitiansebagai berikut: File yang akan digunakan pada proses enkripsi dan dekripsi yaitu berupa file teks berformat rtf dan txt, pada proses pembangunan sistem, tools yang akan digunakan yaitu *Visual Basic.Net* 2010. Penelitian ini akan terfokus pada algoritma *Merkle-Hellman Knapsack* dengan panjang kunci yang dapat digunakan sebanyak 8 Digit

2. METODE PENELITIAN

Analisis Permasalahan

Masalah yang diangkat dari penelitian tugas akhir ini adalah pembuatan sistem pengamanan *file* teks menggunakan algoritma *Merkle-Hellman Knapsack*. Dimana algoritma *Merkle-Hellman Knapsack* merupakan metode yang digunakan untuk

mengenkripsi *file* teks yang berekstensi txt dan rtf. Metode *Merkle-Hellman Knapsack* merupakan kriptosistem yag menggunakan algoritma asimetris. Kelebihan algoritma asimetris ini adalah proses pendistribusian kunci pada media yang tidak aman seperti internet, tidak memerlukan kerahasian. Karena kunci yang didistribusikan adalah kunci publik. Sehingga jika kunci ini sampai hilang atau diketahui oleh orang lain yang tidak berhak, maka pesan sandi yang dikirim akan tetap aman. Sedangkan kunci *private* tetap disimpan (tidak didistribusikan).

Proses Enkripsi Algoritma *Merkle-Hellman Knapsack*

Proses enkripsi algoritma *Merkle-Hellman Knapsack* menggunakan kunci *public* yang diberikan oleh pengguna. Adapun proses enkripsi pada algoritma *Merkle-Hellman Knapsack* adalah sebagai berikut:

Diketahui bilangan super-increasing (s) :

s = {2, 7, 11, 21, 42, 89, 180, 354}

Diketahui *public key* (t) :

218-763-417-725-668-317-70-268

Perhitungan *Public Key* (t) :

t1 = a*s1 mod p = 109 * 2 mod 782 = 218

t2 = a*s2 mod p = 109 * 7 mod 782 = 763

t3 = a*s3 mod p = 109 * 11 mod 782 = 417

t4 = a*s4 mod p = 109 * 21 mod 782 = 725

Proses Deskripsi Algoritma *Merkle-Hellman Knapsack*

Proses dekripsi algoritma *Merkle-Hellman Knapsack* menggunakan kunci *private* yang di-input oleh pengguna. Pada proses deskripsi akan digunakan variable tambahan yang disebut dengan variable $a\ inverse$ yang diperoleh dari proses modulo *inverse* antara variabel a dan variabel p. Adapun proses deskripsi pada algoritma *Merkle-Hellman Knapsack* adalah sebagai berikut:

1. Menghitung nilai modulo *inverse* dari variabel a dan variabel p.

   Diperoleh nilai modulo *inverse* adalah sebesar 617 dikarenakan memenuhi syarat berikut:

   (617 * 109) mod 782 = 1

2. Menghitung nilai *plaintext* sementara.

   P(1) = (1826 * 617) mod 782 = 562

   P(2) = (1805 * 617) mod 782 = 117

   P(3) = (1805 * 617) mod 782 = 117

   P(4) = (1431 * 617) mod 782 = 49

3. Berikutnya adalah mendekomposisikan nilai *plaintext* sementara dengan mengurangi nilai *plaintext* sementara dengan nilai *super-increasing* (s) yang paling dekat dan lebih kecil :

   P(1) :

   562 – 354 = 208 ‘dikurangi dengan 354 dikarenakan paling dekat dan lebih kecil dibandingkan dengan nilai lain dari *super-increasing* (s)

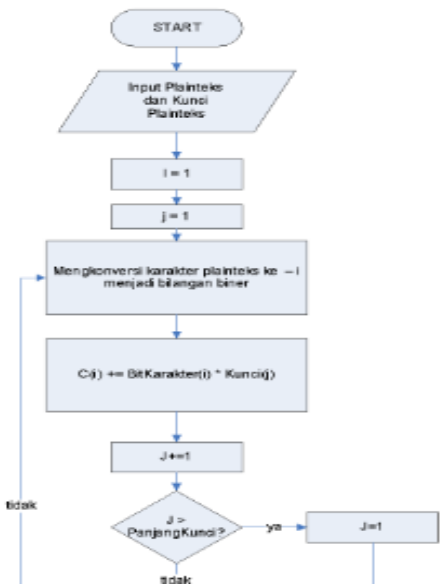
   208 – 180 = 28

   28 – 21 = 7

   7 – 7 = 0

Perancangan Sistem

Pada tahapan perancangan sistem ini akan di jelaskan proses keseluruhan dari program yang akan di buat, agar proses alurnya lebih jelas dan sesuai dengan *standart Unified Modern Language* (UML) yang akan di rancang.



JIKSTRA Vol. 6, No. 01, April 2024 *Halaman. 73-84*
p-ISSN :2715-887X
e-ISSN: 2747-2485

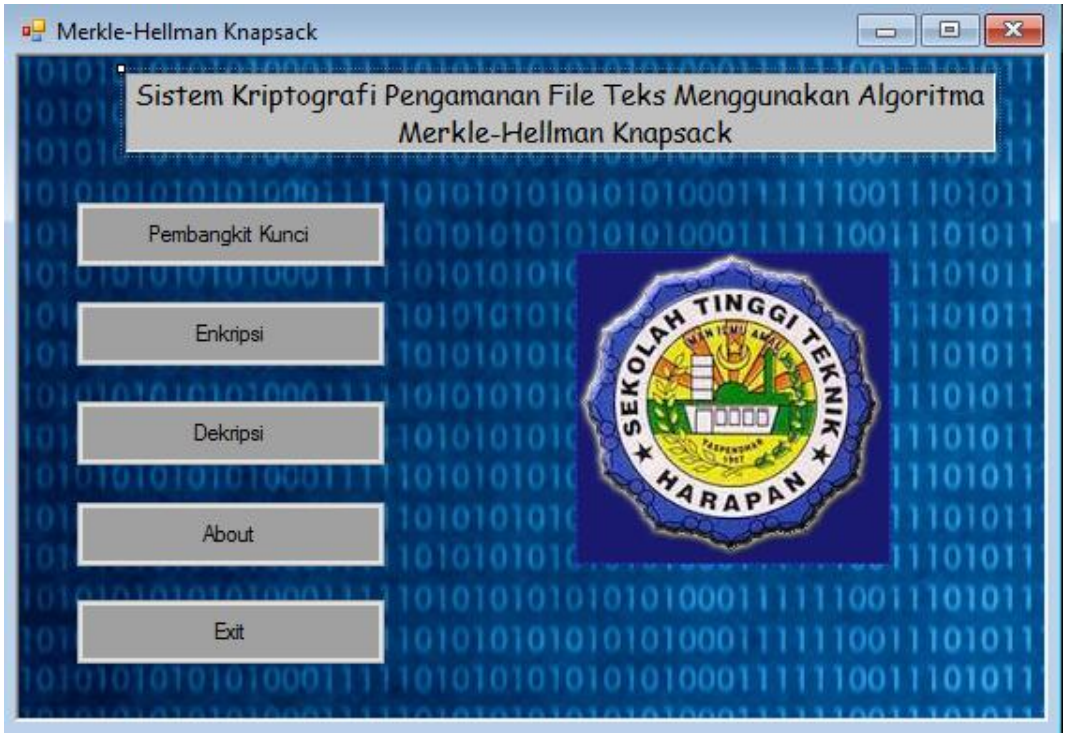
3. HASIL DAN PEMBAHASAN

Gambar 1. Flowchart Proses Enkripsi Algoritma *Merkle-Hellman Knapsack*

3. HASIL DAN PEMBAHASAN

Tampilan Program

Pada tampilan menu utama terdapat lima menu utama yaitu menu pembangkit kunci, enkripsi, dekripsi, *about* dan *exit*. Adapun tampilan dari menu utama dapat dilihat pada gambar 2 berikut.

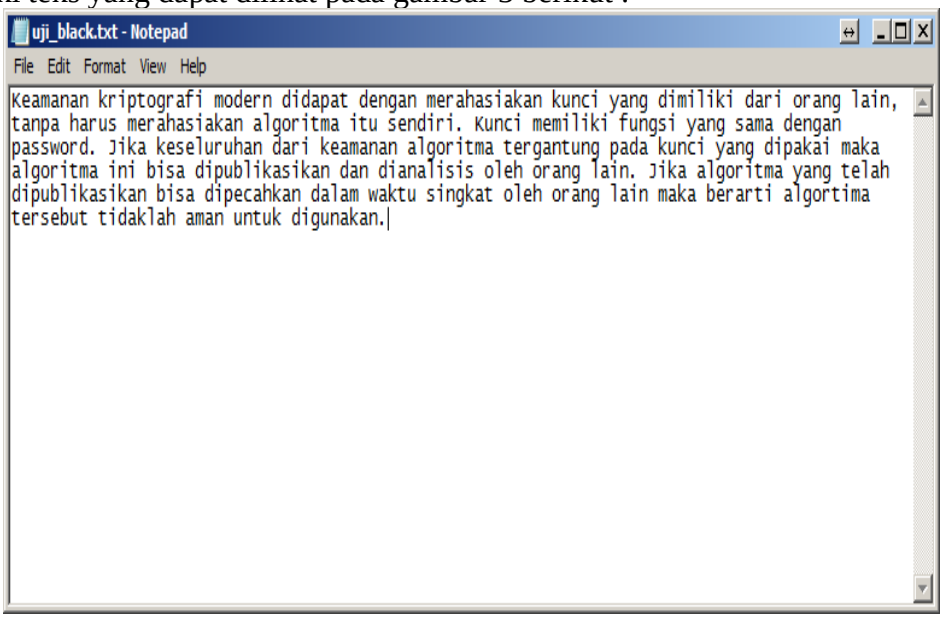


Gambar 2. Tampilan Menu Utama

.

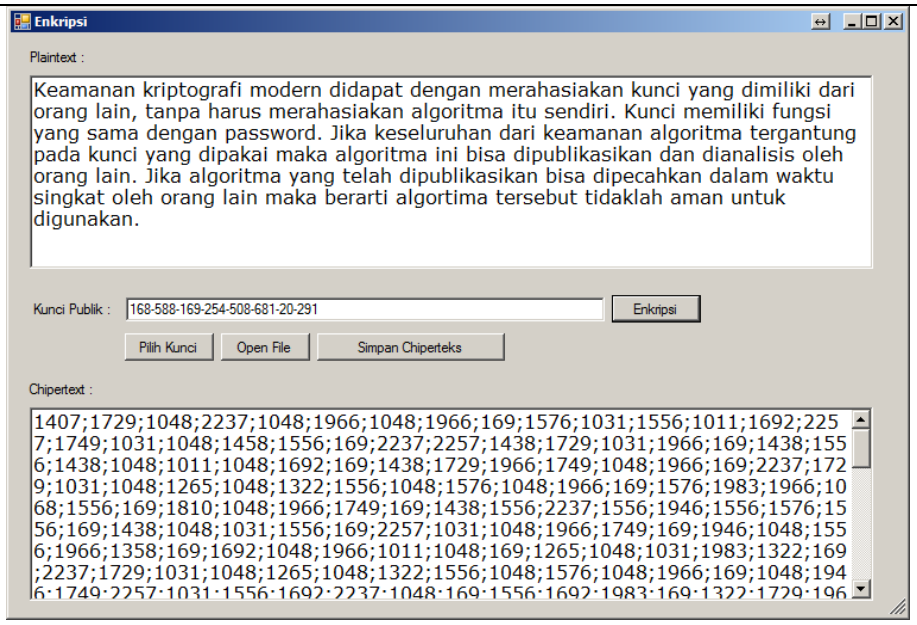
Pengujian *Blackbox*

Pengujian *blackbox* dilakukan untuk memperoleh validasi terhadap fungsionalitas yang dimiliki oleh aplikasi yang dikembangkan. Pengujian *blackbox* pada penelitian ini dilakukan dengan menggunakan *input* teks yang akan dienkripsi dan didekripsi kembali. Input teks yang digunakan pada pengujian ini adalah *input* ”uji_black.txt” yang mana memiliki teks yang dapat dilihat pada gambar 3 berikut :



Gambar 3. Input Pengujian *Blackbox*

Pengujian *blackbox* yang dilakukan pada penelitian ini dimulai dengan proses enkripsi, kemudian dilanjutkan dengan melihat hasil chiperteks setelah proses enkripsi, pengujian terus dilanjutkan dengan proses dekripsi yang akan memeriksa apakah berkas atau teks dapat dikembalikan ke kondisi saat sebelum di enkripsi. Pengujian enkripsi pada pengujian ini yang dapat dilihat pada gambar 4 berikut.



Gambar 4. Pengujian Enkripsi *Blackbox*

Hasil pengujian enkripsi seperti yang terlihat pada gambar 4 dapat dilihat chiperteks hasil enkripsi telah berubah sehingga proses enkripsi berhasil menerapkan metode *Merkle-Hellman* pada plainteks yang digunakan. Adapun parameter yang digunakan pada proses enkripsi adalah :
Kunci Publik: 168-588-169-254-508-681-20-291

Plainteks:
Keamanan kriptografi modern didapat dengan merahasiakan kunci yang dimiliki dari orang lain, tanpa harus merahasiakan algoritma itu sendiri. Kunci memiliki fungsi yang sama dengan password. Jika keseluruhan dari keamanan algoritma tergantung pada kunci yang dipakai maka algoritma ini bisa dipublikasikan dan dianalisis oleh orang lain. Jika algoritma yang telah dipublikasikan bisa dipecahkan dalam waktu singkat oleh orang lain maka berarti algortima tersebut tidaklah aman untuk digunakan.

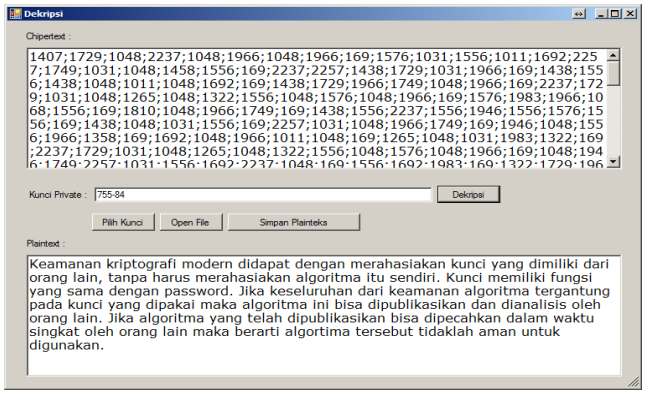
Chiperteks:
1407;1729;1048;2237;1048;1966;1048;1966;169;1576;1031;1556;1011;1692;2257;174
9;1031;1048;1458;1556;169;2237;2257;1438;1729;1031;1966;169;1438;1556;1438;104
8;1011;1048;1692;169;1438;1729;1966;1749;1048;1966;169;2237;1729;1031;1048;126
5;1048;1322;1556;1048;1576;1048;1966;169;1576;1983;1966;1068;1556;169;1810;104
8;1966;1749;169;1438;1556;2237;1556;1946;1556;1576;1556;169;1438;1048;1031;155
6;169;2257;1031;1048;1966;1749;169;1946;1048;1556;1966;1358;169;1692;1048;1966
;1011;1048;169;1265;1048;1031;1983;1322;169;2237;1729;1031;1048;1265;1048;1322
;1556;1048;1576;1048;1966;169;1048;1946;1749;2257;1031;1556;1692;2237;1048;169
;1556;1692;1983;169;1322;1729;1966;1438;1556;1031;1556;1378;169;1407;1983;1966

.

;1068;1556;169;2237;1729;2237;1556;1946;1556;1576;1556;169;1458;1983;1966;1749;1322;1556;169;1810;1048;1966;1749;169;1322;1048;2237;1048;169;1438;1729;1966;1749;1048;1966;169;1011;1048;1322;1322;2003;2257;1031;1438;1378;169;1116;1556;1576;1048;169;1576;1729;1322;1729;1946;1983;1031;1983;1265;1048;1966;169;1438;1048;1031;1556;169;1576;1729;1048;2237;1048;1966;1048;1966;169;1048;1946;1749;2257;1031;1556;1692;2237;1048;169;1692;1729;1031;1749;1048;1966;1692;1983;1966;1749;169;1011;1048;1438;1048;169;1576;1983;1966;1068;1556;169;1810;1048;1966;1749;169;1438;1556;1011;1048;1576;1048;1556;169;2237;1048;1576;1048;169;1048;1946;1749;2257;1031;1556;1692;2237;1048;169;1556;1966;1556;169;777;1556;1322;1048;169;1438;1556;1011;1983;777;1946;1556;1576;1048;1322;1556;1576;1048;1966;169;1438;1048;1966;169;1438;1556;1048;1966;1048;1946;1556;1322;1556;1322;169;2257;1946;1729;1265;169;2257;1031;1048;1966;1749;169;1946;1048;1556;1966;1378;169;1116;1556;1576;1048;169;1048;1946;1749;2257;1031;1556;1692;2237;1048;169;1

810;1048;1966;1749;169;1692;1729;1946;1048;1265;169;1438;1556;1011;1983;777;1946;1556;1576;1048;1322;1556;1576;1048;1966;169;777;1556;1322;1048;169;1438;1556;1011;1729;1068;1048;1265;1576;1048;1966;169;1438;1048;1946;1048;2237;169;2003;1048;1576;1692;1983;169;1322;1556;1966;1749;1576;1048;1692;169;2257;1946;1729;1265;169;2257;1031;1048;1966;1749;169;1946;1048;1556;1966;169;2237;1048;1576;1048;169;777;1729;1031;1048;1031;1692;1556;169;1048;1946;1749;2257;1031;1692;1556;2237;1048;169;1692;1729;1031;1322;1729;777;1983;1692;169;1692;1556;1438;1048;1576;1946;1048;1265;169;1048;2237;1048;1966;169;1983;1966;1692;1983;1576;169;1438;1556;1749;1983;1966;1048;1576;1048;1966;1378

Pengujian selanjutnya dilakukan dengan melakukan dekripsi terhadap chiperteks yang dihasilkan dari proses enkripsi. Pengujian dekripsi pada pengujian ini yang dapat dilihat pada gambar 5 berikut.



Gambar 5. Pengujian Dekripsi *Blackbox*

Pengujian dekripsi seperti yang terlihat pada gambar 5 dapat dilihat proses dekripsi balik terhadap berkas chiperteks yang dienkripsi sebelumnya dapat dikembalikan menjadi berkas teks semula dengan sempurna. Adapun parameter proses dekripsi yang digunakan adalah :

Kunci Private : 755-84
Chiperteks :
1407;1729;1048;2237;1048;1966;1048;1966;169;1576;1031;1556;1011;1692;2257;174
9;1031;1048;1458;1556;169;2237;2257;1438;1729;1031;1966;169;1438;1556;1438;104
8;1011;1048;1692;169;1438;1729;1966;1749;1048;1966;169;2237;1729;1031;1048;126
5;1048;1322;1556;1048;1576;1048;1966;169;1576;1983;1966;1068;1556;169;1810;104
8;1966;1749;169;1438;1556;2237;1556;1946;1556;1576;1556;169;1438;1048;1031;155
6;169;2257;1031;1048;1966;1749;169;1946;1048;1556;1966;1358;169;1692;1048;1966
;1011;1048;169;1265;1048;1031;1983;1322;169;22

37;1729;1031;1048;1265;1048;1322;1556;1048;1576;1048;1966;169;1048;1946;1749;2
257;1031;1556;1692;2237;1048;169;1556;1692;1983;169;1322;1729;1966;1438;1556;1
031;1556;1378;169;1407;1983;1966;1068;1556;169;2237;1729;2237;1556;1946;1556;1
576;1556;169;1458;1983;1966;1749;1322;1556;169;1810;1048;1966;1749;169;1322;10
48;2237;1048;169;1438;1729;1966;1749;1048;1966;169;1011;1048;1322;1322;2003;22
57;1031;1438;1378;169;1116;1556;1576;1048;169;1576;1729;1322;1729;1946;1983;10
31;1983;1265;1048;1966;169;1438;1048;1031;1556;169;1576;1729;1048;2237;1048;19
66;1048;1966;169;1048;1946;1749;2257;1031;1556;1692;2237;1048;169;1692;1729;10
31;1749;1048;1966;1692;1983;1966;1749;169;1011;1048;1438;1048;169;1576;1983;19
66;1068;1556;169;1810;1048;1966;1749;169;1438;1556;1011;1048;1576;1048;1556;16
9;2237;1048;1576;1048;169;1048;1946;1749;2257;1031;1556;1692;2237;1048;169;155
6;1966;1556;169;777;1556;1322;1048;169;1438;1556;1011;1983;777;1946;1556;1576;
1048;1322;1556;1576;1048;1966;169;1438;1048;1966;169;1438;1556;1
048;1966;1048;1946;1556;1322;1556;1322;169;2257;1946;1729;1265;169;2257;1031;1
048;1966;1749;169;1946;1048;1556;1966;1378;169;1116;1556;1576;1048;169;1048;19
46;1749;2257;1031;1556;1692;2237;1048;169;1810;1048;1966;1749;169;1692;1729;19
46;1048;1265;169;1438;1556;1011;1983;777;1946;1556;1576;1048;1322;1556;1576;10
48;1966;169;777;1556;1322;1048;169;1438;1556;1011;1729;1068;1048;1265;1576;104
8;1966;169;1438;1048;1946;1048;2237;169;2003;1048;1576;1692;1983;169;1322;1556
;1966;1749;1576;1048;1692;169;2257;1946;1729;1265;169;2257;1031;1048;1966;1749
;169;1946;1048;1556;1966;169;2237;1048;1576;1048;169;777;1729;1031;1048;1031;1
692;1556;169;1048;1946;1749;2257;1031;1692;1556;2237;1048;169;1692;1729;1031;1
322;1729;777;1983;1692;169;1692;1556;1438;1048;1576;1946;1048;1265;169;1048;22
37;1048;1966;169;1983;1966;1692;1983;1576;169;1438;1556;1749;1983;1966;1048;15
76;1048;1966;1378

Plainteks :
Keamanan kriptografi modern didapat dengan merahasiakan kunci yang dimiliki dari orang lain, tanpa harus merahasiakan algoritma itu sendiri. Kunci memiliki fungsi yang sama dengan password. Jika keseluruhan dari keamanan algoritma tergantung pada kunci yang dipakai maka algoritma ini bisa dipublikasikan dan dianalisis oleh orang lain. Jika algoritma yang telah dipublikasikan bisa dipecahkan dalam waktu singkat oleh orang lain maka berarti algortima tersebut tidaklah aman untuk digunakan.

Pengujian *blackbox* yang dilakukan menunjukkan bahwa aplikasi yang dikembangkan memiliki fungsi yang sesuai dan dapat beroperasi sesuai dengan yang diharapkan baik pada saat proses enkripsi maupun pada saat proses dekripsi.

4. KESIMPULAN

Berdasarkan proses perancangan kriptografi menggunakan metode *Merkle-Hellman Knapsack* maka dapat diambil beberapa kesimpulan yaitu:

1. Aplikasi ini menerapkan algoritma Merkle-Hellman Knapsack dalam proses enkripsi dan dekripsi pada *file.*
2. Aplikasi ini terdiri dari tiga komponen utama yaitu komponen pembangkit kunci, enkripsi dan dekripsi.
3. *File* yang digunakan pada proses enkripsi dan dekripsi ini berformat rtf dan txt.

DAFTAR PUSTAKA

[1] Adi Nugroho. 2010. *Rekayasa Perangkat Lunak Berbasis Objek dengan Metode USDP*. Andi. Yogyakarta.

[2] Afyenni, R. 2014. *Perancangan Data Flow Diagram Untuk Sistem Informasi Sekolah (Studi Kasus SMA Pembangungn Laboratorium UNP)*. TEKNOIF , 35-39.

[3] Agarwal, A. 2011. *Encrypting Messages using the Merkle-Hellman Knapsack Cryptosystem.* International Journal of Computer Science and Network Security , 12-14.

[4] Akik Hidayat, A. R. 2016. *Cryptography Asymmetries Merkle-Hellman Knapsack Digunakan Untuk Enkripsi Dan Dekripsi Teks. Seminar Nasional MIPA*, (pp. 66-68). Jatinangor.

[5] Chandra. 2016. *Keamanan Data Dengen Kriptografi Kunci Publik.* times , 11-15.

[6] Dharwiyanti. 2003. *Pengantar Unified Modelling Language (UML), www.ilmukomputer.com.*

[7] Fresly, I. A. 2015. *Implementasi Kriptgrafi Pengamanan Data Pada pesan Teks, Isi File Dokumen dan file Dokumen Menggunakan Algoritma Advanced Encryption Standard.* Informatika Mulawarman , 20-31.

[8] Kaluge, G. R. 2013. *Penambahan Permutasi Pada Knapsack Chiper.* Program Studi Teknik Elektro Dan informatika .

[9] Muslihudin, Muhamad Oktafianto. (2016). *Analisis dan Perancangan Sistem Informasi Menggunakan Model Terstruktur dan UML*. Yogyakarta.

[10] Muslim, A. 2013. *Membangun Aplikasi Autogenerate Scrpt Ke Flowchart Untuk Mendukung Business Process Reengineering*. Sarjana Teknik Informatika , 448-456.

[11] Nuraini, Rini. 2015. *Desain Algorithma Operasi Perkalian Matriks Menggunakan Metode Flowchart*. Februari 2015 Vol. 1 No.1. Jakarta. AMIK BSI.

[12] Opik, A. 2013. *Pembuatan Aplikasi Anbiyapedia Ensiklopedi Muslim Anak Berbasis Web*. Teknik Informatika . Patricia handoko, A. 2015. *Teknik Keamanan Data Menggunakan Kriptografi Dengan Algoritma Vigenere Chiper Dan Steganografi*

[13] *Dengan Metode End Of File(EOF)*. Program Studi Teknik Informatika Fakultas Ilmu Komputer Universitas Dian Nuswantoro .

[14] Purwadi, H. j. 2014. *Aplikasi Kriptografi Asimetris Dengan Metode Diffie-Hellman Dan Algoritma Elgamal Untuk Keamanan Teks*. SAINTIKOM , 183-196.

[15] Sugianto,. 2014. *Kriptografi Gabungan Menggunakan Algoritma Mono Alphabetic Dan One Time Pad*. Januari 2014 Vol. 4 No.1. Pontianak. Sekolah Tinggi Manajemen Informatika dan Komputer.