Jurnal Teknologi Informasi dan Ilmu Komputer (JTIIK) DOI: 10.25126/jtiik.201744468
Vol. 4, No. 4, Desember 2017, hlm. 268-274 p-ISSN: 2355-7699
Akreditasi KEMENRISTEKDIKTI, No. 51/E/KPT/2017 e-ISSN: 2528-6579

REKAYASA APLIKASI KRIPTOGRAFI DENGAN PENERAPAN KOMBINASI ALGORITMA KNAPSACK MERKLE HELLMAN DAN AFFINE CIPHER

Muhammad Fadlan[1], Hadriansa[2]

[1]Sistem Informasi, STMIK PPKIA Tarakanita Rahmawati, Kota Tarakan, Indonesia
[2]Teknik Informatika, STMIK PPKIA Tarakanita Rahmawati, Kota Tarakan, Indonesia
Email: [1]thecuexman@gmail.com, [2]ansar@ppkia.ac.id

(Naskah masuk: 25 September 2017, diterima untuk diterbitkan: 24 Desember 2017)

Abstrak

Kerahasiaan sebuah data merupakan hal yang sangat penting untuk dijaga. Tetapi masalah yang terkadang muncul adalah adanya data rahasia yang bocor atau dicuri oleh pihak-pihak tertentu yang tidak berwewenang. Hal ini dikarenakan tidak adanya pengamanan untuk data tersebut dan diperkuat lagi dengan semakin sering dan mudahnya data untuk didistribusikan melalui sebuah pesan elektronik. Ketika pesan itu diretas, maka data rahasia yang belum diamankan dapat dengan mudah diketahui oleh pihak yang tidak berwewenang. Kriptografi mampu menjadi salah satu cara untuk mengamankan sebuah data. Pada penelitian ini akan dilakukan kombinasi Metode Affine Cipher dan Knapsack Merkle Hellman untuk melakukan enkripsi dan dekripsi data teks. Awalnya data akan dienkripsi terlebih dahulu dengan menggunakan affine cipher, hasil enkripsi dari affine cipher kemudian akan dienkripsi lagi dengan menggunakan merkle hellman. Sedangkan, untuk tahap dekripsi, data akan didekripsi terlebih dahulu menggunakan merkle hellman, kemudian dilanjutkan dengan affine cipher. Kombinasi metode dilakukan untuk lebih memperkuat pengamanan terhadap sebuah data. Hasilnya, kriptografi dengan menggunakan kombinasi metode Affine Cipher dan Knapsack Merkle Hellman dapat dilakukan untuk melakukan enkripsi dan dekripsi terhadap sebuah data. Dengan adanya pengamanan terhadap data tersebut, maka akan menyulitkan dan memperkecil kemungkinan pihak-pihak yang tidak berkepentingan untuk mengetahui suatu data rahasia.

Kata kunci: *affine, dekripsi, enkripsi, knapsack merkle hellman, kriptografi, kerahasiaan*

Abstract

Confidentiality of data is very important to be maintained. But the problem that sometimes arises is the existence of secret data that is leaked or stolen by certain unauthorized parties. This is because there is no security for the data and is reinforced by the more frequent and easy data to be distributed via an electronic message. When the message is hacked, the secret data that has not been secured can easily be known by unauthorized parties. Cryptography can be one way to secure a data. In this research will be a combination of Affine Cipher Method and Knapsack Merkle Hellman to encrypt and decrypt text data. Initially the data will be encrypted first by using affine cipher, the encryption of the affine cipher will then be encrypted again using merkle hellman. Meanwhile, for the decryption stage, the data will be decrypted first using merkle hellman, then continued with affine cipher. Combination methods are done to further strengthen the security of a data. The result, cryptography by using a combination of Affine Cipher and Knapsack Merkle Hellman methods successfully done to perform encryption and decryption of a data. With the security of the data, it will complicate and minimize the possibility of parties who are not interested to know a secret data.

Keywords: *affine, decryption, encryption, knapsack merkle hellman, cryptography, confidentiality*

1. PENDAHULUAN

Menjaga kerahasiaan sebuah data, agar tidak mudah jatuh kepada pihak yang tidak berwenang merupakan hal yang sangat penting. Bayangkan saja suatu data rahasia yang bahkan dapat bersifat pribadi dengan mudah bocor bahkan diketahui oleh pihak-pihak yang tidak berkepentingan, tentu saja menimbulkan dampak yang sangat merugikan bagi pihak-pihak yang memiliki data tersebut. Masalah keamanan data pada komputer khususnya, menjadi isu yang sangat penting pada era digital saat ini. Para pelaku kejahatan cyber biasanya memanfaatkan celah-celah keamanan agar dapat memasuki, mengambil bahkan memanipulasi / mengubah data.

Selain itu, perkembangan yang pesat dalam proses pendistribusian data juga membawa dampak yang besar yaitu masalah keamanan dari suatu data. Hal tersebut dapat membuat para pemilik data menginginkan sesuatu yang lebih aman yang dapat melindungi suatu data yang bersifat rahasia tersebut. Untuk itu diperlukan suatu teknik dalam menjaga

kerahasiaan dari sebuah data. Teknik yang dimaksud adalah Ilmu Kriptografi. Dengan menggunakan kriptografi maka suatu data dapat diamankan dengan mengaburkan / mengubah / mengacak isi dari suatu data melalui proses enkripsi. Data tersebut tentu saja dapat dikembalikan ke bentuk semula melalui proses dekripsi. Keuntungan dalam menggunakan kriptografi adalah hanya pihak berwenang terhadap data saja yang dapat melakukan tahap enkripsi maupun dekripsi. Sehingga pihak-pihak yang tidak berwenang sulit untuk mengetahui isi dari data yang telah diacak / dienkripsi.

Awalnya, kriptografi hanya dipahami sebagai ilmu yang digunakan untuk menyembunyikan pesan. Namun, kini telah bergeser seiring dengan perkembangan jaman menjadi ilmu yang terkait dengan teknik matematika yang digunakan dalam keamanan informasi seperti keutuhan dan kerahasiaan data, serta pengesahan entitas (Sadikin, 2012).

Terdapat beberapa metode kriptografi ataupun algoritma dalam melakukan enkripsi dan dekripsi data. Di beberapa penelitian ada yang hanya menggunakan satu metode saja dalam pengamanan data. Namun pada penelitian ini akan menggunakan kombinasi dari dua metode dalam melakukan pengamanan data. Pemilihan dua metode dilakukan untuk meningkatkan keamanan dari sebuah data serta semakin menyulitkan bagi pihak-pihak yang tidak berkepentingan untuk mengetahui sebuah data.

Data yang diujicoba pada penelitian ini adalah berupa teks dan metode yang digunakan adalah kombinasi dari metode *Affine Cipher* dan *Knapsack Merkle Hellman*. Metode Affine cipher digunakan karena efisien dan efektif untuk mengamankan data, informasi, maupun dokumen-dokumen penting sehingga tidak dapat disalahgunakan oleh pihak yang tidak bertanggung jawab (Babu, 2017). Sedangkan kelebihan dari merkle hellman adalah tidak diperlukannya kerahasiaan pada proses pendistribusian key. Hal ini dikarenakan key yang disalurkan / dibagikan berupa *public key*. Meskipun kunci ini diketahui oleh orang lain yang tidak berwenang, maka pesan akan tetap terjaga kerahasiaannya. Sedangkan *private key* akan tetap disimpan atau tidak didistribusikan (Hidayat, Akmal, & Rosyadi, 2016). Dengan kombinasi dari dua metode tersebut diharapkan dapat lebih memperkuat pengamanan terhadap sebuah pesan / data dibandingkan dengan menggunakan satu metode.

2. TINJAUAN UMUM

2.1. Keamanan Data

Masalah yang sangat penting pada era digital dan komputerisasi pada saat ini adalah masalah keamanan dan kerahasiaan data. Keamanan sebuah data sangat perlu untuk diperhatikan, mengingat semakin banyaknya pelaku-pelaku kejahatan *cyber*. Sebuah data yang dianggap rahasia kebanyakan berupa data-data yang dianggap penting dan terbatas hanya boleh diketahui oleh orang-orang tertentu.

Banyaknya keuntungan atau dampak positif dari kemajuan teknologi, juga diikuti oleh banyaknya dampak negatif yang ditimbulkan, seperti *computer crime* yang terdiri atas pemerasan, pencurian dan penipuan, serta kompetisi. Jatuhnya data kepihak lain tentu saja dapat menimbulkan kerugian bagi pemilik data / informasi. (Ariyus, 2008).

Keamanan jaringan menjadi lebih penting karena jumlah data yang dipertukarkan di Internet meningkat dengan drastis. Oleh karena itu, kerahasiaan dan integritas data diperlukan untuk melindungi dari akses yang tidak sah / pihak-pihak yang tidak berwenang (Moon & Kawitkar, 2007). Keamanan dan kerahasiaan data pada jaringan komputer saat ini menjadi isu yang sangat penting dan terus berkembang. Komputer sebagai sarana untuk menyimpan dan mentransmisikan data, informasi, dan dokumen rahasia, seringkali mudah diakses oleh orang yang tidak bertanggung jawab (Babu, 2017).

Oleh karena itu, pengamanan sebuah data kiranya sangat penting dan dibutuhkan, agar data / informasi yang dimiliki khususnya informasi yang bersifat rahasia dapat tetap terjaga kerahasiaannya, walaupun data tersebut telah jatuh kepihak yang tidak berhak / berkepentingan.

2.2. Kriptografi

Salah satu solusi yang dapat digunakan untuk menjamin kerahasiaan maupun keamanan dari suatu informasi adalah dengan kriptografi. Dengan kriptografi sebuah informasi dapat diacak atau disandikan menjadi informasi yang sulit atau bahkan tidak dipahami melalui sebuah proses yang dinamakan dengan enkripsi (Murdani, 2017). Pada kriptografi juga akan dipelajari teknik-teknik matematika yang terkait dengan aspek keamanan informasi (Munir, 2006). Sebenarnya, teknik menjaga kerahasiaan pesan tidak hanya dengan menggunakan kriptografi. Ada juga teknik lain yang dapat digunakan yaitu steganografi. Steganografi sangat kontras dengan kriptografi (Fadlan & Deby, 2014).

Kriptografi memegang peranan penting dalam memberikan keamanan terhadap data yang dikirimkan melalui internet (Ali, 2014). Dengan kriptografi, maka pengamanan sebuah data dapat dilakukan. Suatu data yang tadinya bisa dibaca / dikenali dengan mudah, maka dengan kriptografi akan menjadi sulit dikenali karena telah melalui proses pengacakan pada tahap enkripsi. Pada kriptografi terdapat 2 tahap yang paling utama, yaitu enkripsi dan dekripsi. Pada tahap enkripsi, akan dilakukan pengacakan sebuah data / teks ke dalam format atau bentuk yang susah untuk dikenali *(cipherteks)*. Sedangkan tahap dekripsi adalah

tahapan untuk mengubah data yang telah diacak ke dalam bentuk aslinya (*plainteks*).

Akhir-akhir ini, persaingan dalam menggunakan kriptografi semakin berkembang, banyak individu maupun organisasi yang telah menggunakan kriptografi untuk mengamankan data atau pesan yang mereka punya, sehingga data tersebut tidak dapat diketahui oleh pihak yang tidak berwenang (Handayani, Pratitis, Nur, Mashuri, & Nugroho, 2017).

2.3. *Affine* dan *Knapsack Merkle Hellman*

Affine Cipher merupakan pengembangan dari Caesar Cipher. Pada affine, plainteks akan dikalikan dengan sebuah nilai dan ditambahkan dengan sebuah pergeseran. Sedangkan Knapsack adalah algoritma kriptografi publik yang keamanannya terletak pada sulitnya memecahkan persoalan knapsack (Munir, 2006)

Gagasan tentang sebuah affine cipher adalah menggunakan perkalian dikombinasikan dengan penambahan, modulo m, di mana m adalah bilangan bulat, untuk membuat substitusi campuran. Secara umum, affine adalah sistem cipher dimana huruf plainteks dienkripsikan secara matematis (Mokhtari & Hasan, 2012)

Metode Knapsack Merkle Hellman telah banyak digunakan untuk memodelkan solusi masalah di industri seperti pada kriptografi kunci publik. Knapsack Merkle-Hellman merupakan metode dalam kriptografi yang menggunakan algoritma asimetris dan memiliki 2 kunci utama, yakni kunci publik dan kunci privat. Kunci yang didistribusikan dikenal dengan istilah kunci publik, jika kunci publik ini diketahui oleh orang lain yang tidak berhak / berkepentingan, maka data yang dikirim akan tetap aman. Untuk kunci private adalah kunci yang tetap disimpan oleh pihak-pihak yang berhak.

Ide dasar di balik skema enkripsi Merkle-Hellman adalah menciptakan masalah subset yang bisa dipecahkan dengan mudah dan kemudian menyembunyikan sifat *superincreasing* dengan perkalian modular dan permutasi. Vektor yang ditransformasikan membentuk pesan terenkripsi dan vektor superincreasing asli membentuk kunci pribadi dan digunakan untuk menguraikan pesan (Agarwal, 2011).

Pada algoritma Merkle-Hellman Knapsack digunakan kunci privat dan kunci publik dalam melakukan proses kriptografinya, metode ini juga memiliki pengamanan ganda sehingga susah untuk ditembus (Murdani, 2017).

3. METODOLOGI PENELITIAN

Dengan melakukan studi literatur terhadap beberapa sumber baik yang berasal dari buku maupun jurnal-jurnal ilmiah nasional maupun internasional terkait dengan masalah pengamanan data, maka pada penelitian ini dilakukan studi kasus untuk mengamankan sebuah data, khususnya data berupa teks dengan membangun sebuah aplikasi berbasis desktop, pengamanan data dilakukan dengan menggunakan metode-metode dalam kriptografi yaitu Affine Cipher dan Knapsack Merkle Hellman. Kombinasi dua metode dilakukan didalam penelitian ini. Metode affine cipher digunakan untuk melakukan enkripsi plainteks tahap pertama sebelum dilakukan enkripsi tahap kedua menggunakan Knapsack Merkle Hellman. Enkripsi dengan metode affine cipher dapat dinyatakan dengan Persamaan 1:

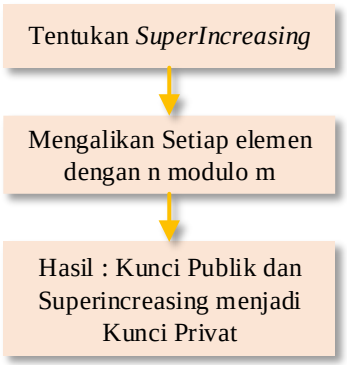
$$C = mP + b \ (mod\ n) \quad (1)$$

Sedangkan, untuk dekripsi pada metode affine digunakan Persamaan 2:

$$P = m^{-1}(C-b) \ (mod\ n) \quad (2)$$

Dimana, C adalah *ciphertext*, P adalah *plaintext*, bilangan bulat yang relatif prima dengan *n* diwakili dengan variabel *m*, *n* adalah ukuran alphabet, dan *b* adalah jumlah pergeseran.

Metode Knapsack Merkle-Hellman digunakan pada tahapan untuk mengenkripsi cipherteks atau hasil enkripsi menggunakan affine cipher dan juga digunakan pada tahapan untuk mendekripsi cipherteks menjadi plainteks yang akan didekripsi lagi menggunakan affine cipher. Hal pertama yang perlu dilakukan pada metode Knapsack Merkle Hellman adalah dengan membangkitkan kunci publik dan privat. Adapun algoritma untuk membangkitkan kunci tersebut dapat terlihat pada Gambar 1.

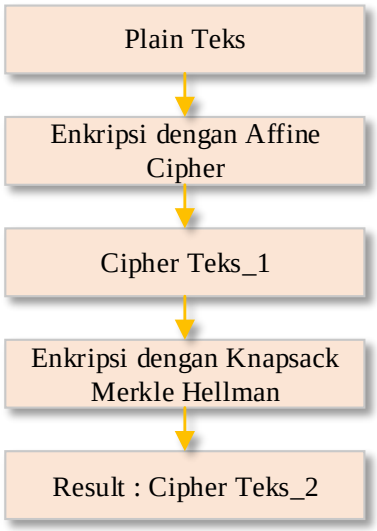


Gambar 1. Algoritma Pembangkitan Kunci

Kunci publik adalah deret angka yang tidak termasuk dalam *superincreasing*, sedangkan kunci privat merupakan deret angka *superincreasing* itu sendiri. Dalam suatu deret angka *superincreasing* terdiri atas 8 angka, yang mana sebuah angka memiliki nilai yang lebih besar dibandingkan dengan penjumlahan semua angka sebelumnya.

Pada Gambar 1 terlihat bahwa algoritma untuk membangkitkan kunci publik dan kunci privat diawali dengan penentuan barisan superincreasing. Yang dilanjutkan dengan pengalian setiap elemen didalam barisan superincreasing yang telah

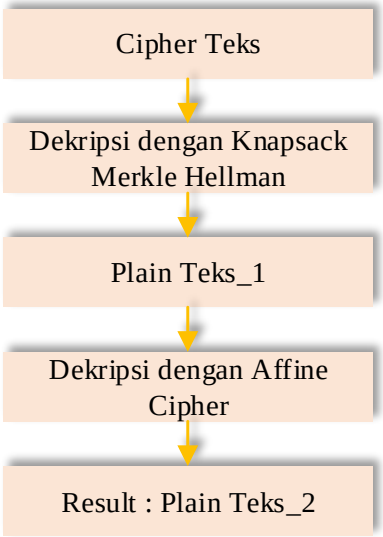
ditentukan dengan *n modulo m* yang merupakan angka yang nilainya lebih besar dibandingkan jumlah seluruh elemen pada suatu deret angka, dan yang terakhir, hasil perkalian tersebut akan menjadi deretan kunci public. Untuk deretan angka superincreasing dianggap sebagai kunci privat.



Gambar 2. Tahapan Enkripsi

Untuk tahapan enkripsi plainteks dengan menggunakan kombinasi dari metode affine cipher dan knapsack merkle-hellman dapat terlihat pada Gambar 2.

Pada Gambar 2 tersebut, dapat dilihat bahwa tahap enkripsi diawali dengan menyiapkan plainteks yang akan kita enkripsi. Enkripsi tahap pertama dimulai dengan menggunakan affine cipher, dimana setelah melalui proses enkripsi affine tersebut, akan menghasilkan sebuah cipher teks pertama (cipher teks_1). Cipher Teks_1 tersebut akan dienkripsi lagi untuk kedua kalinya dengan menggunakan Knapsack Merkle Hellman. Hasil akhir dari dua kali proses enkripsi tersebut berupa cipher teks (cipher teks_2).

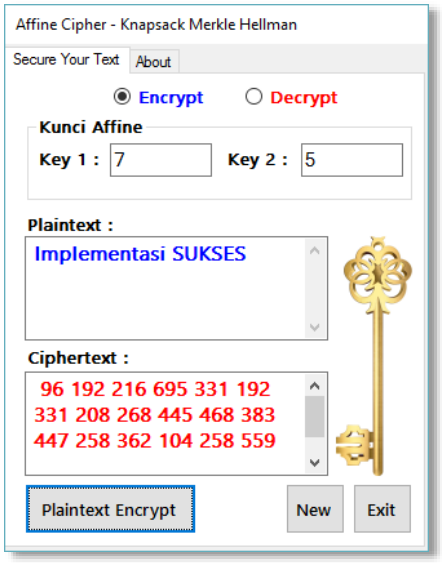


Gambar 3. Tahapan Dekripsi

Proses dekripsi dapat terlihat pada Gambar 3. Dekripsi diawali dengan menyiapkan cipher teks yang akan didekripsi. Kemudian dilanjutkan dengan dekripsi tahap pertama menggunakan Knapsack Merkle Hellman, yang akan menghasilkan plainteks tahap pertama (plain teks_1). Plainteks tersebut bukanlah hasil akhir yang diinginkan. Untuk itu perlu dilakukan dekripsi tahap kedua menggunakan affine cipher. Setelah melalui dekripsi tahap kedua tersebut, maka akan dihasilkan hasil akhir / plain teks yang diinginkan (bentuk semula).

4. HASIL DAN PEMBAHASAN

Setelah melalui tahap perancangan aplikasi, dihasilkan sebuah aplikasi kriptografi dengan penerapan affine cipher dan knapsack merkle hellman, seperti yang terdapat pada Gambar 4, dan 5. Aplikasi tersebut dibuat dengan menggunakan *Microsoft Visual Studio 2012*, dengan nama project *KMHapp*. Pada Gambar 4, merupakan tampilan form ketika melakukan proses enkripsi data. Proses tersebut diawali dengan memilih sesuai kebutuhan pilihan enkripsi atau dekripsi data. Untuk melakukan enkripsi maka dipilih *radiobutton Encrypt*. Kemudian, menginputkan kunci untuk affine cipher. Selanjutnya menginput teks yang akan dienkripsi pada *textbox* plainteks. Yang terakhir dengan menekan tombol *Plaintext Encrypt* untuk melakukan enkripsi. Untuk alur lebih jelas dapat dilihat pada Gambar 7.



Gambar 4. Secure Your Teks (Encrypt)

Seperti contoh huruf ‘S’ yang dapat dilihat pada Gambar 4 tersebut, setelah dilakukan enkripsi menjadi angka ‘258’. Bagi pihak-pihak yang tidak mengetahuinya mungkin saja mereka hanya berpikir bahwa angka ‘258’ itu berupa angka biasa saja, yang tidak memiliki makna. Disisi yang lain, angka ‘258’ tersebut sebenarnya merupakan hasil enkripsi dari karakter ‘S’ menggunakan kombinasi affine dan

knapsack merkle hellman. Karakter 'S' memiliki nilai ascii sebesar 83. Dengan menggunakan key affine 1 = 7 dan key 2 = 5, maka dengan menggunakan persamaan 1, didapat ((83*7) + 5) mod 255 = 76 = L.

L (ascii = 76) tersebut merupakan hasil enkripsi tahap pertama yang akan dilanjutkan dengan knapsack merkle hellman. Deretan *superincreasing* yang digunakan adalah 1,2,4,8,16,32,64,128. Karakter L (ascii = 76) dikonversi menjadi bilangan biner, sehingga didapatkan '01001100'. Nilai tiap-tiap bit pada bilangan biner tersebut dikalikan dengan deretan *public key* (41, 82, 164, 72, 144, 32, 64, 128) yang didapat melalui perhitungan terhadap superincreasingly, sehingga akan ditemukan hasil akhir berupa angka 258.

Contoh perhitungan dapat dilihat pada Tabel 1 dan 2. Pada Tabel 1, memperlihatkan perhitungan enkripsi pada affine cipher, yang mana hasil akhirnya akan dienkripsi lagi menggunakan merkle hellman, yang contoh perhitungannya untuk salah satu karakter dapat dilihat pada Tabel 2.

Tabel 1. Perhitungan enkripsi affine cipher

Karakter	$C = mP + b \pmod n$
I (ascii = 73)	((73*7) + 5) mod 255 = 6
m (ascii = 109)	((109*7) + 5) mod 255 = 3
p (ascii = 112)	((112*7) + 5) mod 255 = 24
l (ascii = 108)	((108*7) + 5) mod 255 = 251
e (ascii = 101)	((101*7) + 5) mod 255 = 202
m (ascii = 109)	((109*7) + 5) mod 255 = 3
e (ascii = 101)	((101*7) + 5) mod 255 = 202
n (ascii = 110)	((119*7) + 5) mod 255 = 10
t (ascii = 116)	((116*7) + 5) mod 255 = 52
a (ascii = 97)	((97*7) + 5) mod 255 = 174
s (ascii = 115)	((115*7) + 5) mod 255 = 45
i (ascii = 105)	((105*7) + 5) mod 255 = 230
S (ascii = 83)	((83*7) + 5) mod 255 = 76
U (ascii = 85)	((85*7) + 5) mod 255 = 90
K (ascii = 75)	((75*7) + 5) mod 255 = 20
S (ascii = 83)	((83*7) + 5) mod 255 = 76
E (ascii = 69)	((83*7) + 5) mod 255 = 233
S (ascii = 83)	((83*7) + 5) mod 255 = 76

Tabel 2. Perhitungan enkripsi merkle hellman

Karakter	C	Biner	Public Key	Hasil
S	76	0	41	0*41 = 0
		1	82	1*82 = 82
		0	164	0*164 = 0
		0	72	0*72 = 0
		1	144	1*144 = 144
		1	32	1*32 = 32
		0	64	0*64 = 0
		0	128	0*128 = 0
		Hasil Akhir		82+144+32 = **258**

Hasil akhir dari tahap enkripsi berupa deretan angka yang dapat terlihat pada Gambar 4, *textbox ciphertext.* Pada Gambar 5, menunjukkan proses dalam melakukan dekripsi. Dalam melakukan dekripsi maka yang dipilih adalah *radiobutton Decrypt*, kemudian dilanjutkan dengan memasukkan kunci untuk affine cipher. Kunci ini harus sama seperti kunci ketika melakukan enkripsi data. Jika tidak sama, maka hasil dekripsi tidak sesuai dengan yang diinginkan atau tidak kembali kebentuk semula. Selanjutnya menginput teks yang akan didekripsi pada *textbox* cipherteks. Untuk melakukan proses dekripsi dapat dilakukan dengan menekan tombol *Cipherteks Decrypt*. Perhitungan dekripsi dapat dilihat pada Tabel 3 dan 4.

Tabel 3. Perhitungan dekripsi merkle hellman

Cipher (c)	$C*r^{-1} \bmod q$	P	P_1
96	96*25 mod 256	96	6
192	192*25 mod 256	192	3
216	216*25 mod 256	24	24
695	695*25 mod 256	223	251
331	331*25 mod 256	83	202
192	192*25 mod 256	192	3
331	331*25 mod 256	83	202
208	208*25 mod 256	80	10
268	268*25 mod 256	44	52
445	445*25 mod 256	117	174
468	468*25 mod 256	180	45
383	383*25 mod 256	103	230
258	258*25 mod 256	50	76
362	362*25 mod 256	90	90
104	104*25 mod 256	40	20
258	258*25 mod 256	50	76
559	559*25 mod 256	151	233
258	258*25 mod 256	50	76

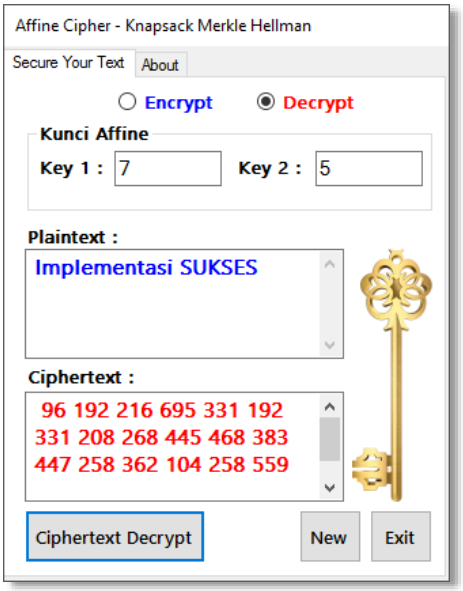
Pada tabel 3 tersebut, terlihat bahwa cipherteks hasil enkripsi akan dikalikan dengan r^{-1} yang merupakan *modulo invers* dari r. Hasil perkalian kemudian dikalikan dengan q yang merupakan angka yang jumlahnya lebih besar dari keseluruhan *superincreasing key* jika di jumlahkan, sehingga akan menghasilkan nilai P dan P_1. Nilai P_1 ini kemudian dianggap sebagai cipher teks pada tahap dekripsi menggunakan affine cipher. Oleh karena itu, nilai tersebut akan didekripsi lagi dengan menggunakan affine cipher. Seperti contoh yang dapat terlihat pada Tabel 4.

Pada Tabel 4, terlihat bahwa nilai P_1 pada dekripsi merkle hellman menjadi nilai C pada dekripsi affine. Setiap nilai akan didekrispsi menggunakan persamaan 2. Contohnya: nilai C = 6, akan dikurangi dengan key 2 (=5) kemudian dikali dengan modulo invers dari m (m^{-1}) yaitu dengan nilai sebesar 73, hasilnya kemudian akan di mod dengan n (=255) sehingga akan menghasilkan nilai P=73. Nilai P=73 tersebut kemudian dikonversi lagi untuk mengetahui karakter apakah yang memiliki nilai 73, setelah dikonversi hasilnya menjadi karakter I. Proses tersebut kemudian diulangi terus-menerus sampai semua nilai C didekripsi.

Tabel 4. Perhitungan dekripsi affine

C	m^{-1} *(C-b) (mod n)*	P	Convert
6	73(6-5) mod 255	73	I
3	73(3-5) mod 255	109	m
24	73(24-5) mod 255	112	p
251	73(251-5) mod 255	108	l
202	73(202-5) mod 255	101	e
3	73(3-5) mod 255	109	m
202	73(202-5) mod 255	101	e
10	73(10-5) mod 255	110	n
52	73(52-5) mod 255	116	t
174	73(174-5) mod 255	97	a
45	73(45-5) mod 255	115	s
230	73(230-5) mod 255	105	i
76	73(76-5) mod 255	83	S
90	73(90-5) mod 255	85	U
20	73(20-5) mod 255	75	K
76	73(76-5) mod 255	83	S
233	73(233-5) mod 255	69	E
76	73(76-5) mod 255	83	S

Sedangkan untuk implementasi pada program, maka hasil dari dekripsi tersebut akan terlihat seperti Gambar 5 dibawah ini. Jadi, ketika cipher teks dimasukkan pada *textbox* cipherteks, memasukkan kunci affine 1 dan 2, dan menekan tombol ciphertext maka hasilnya akan tampil pada *textbox plaintext.*



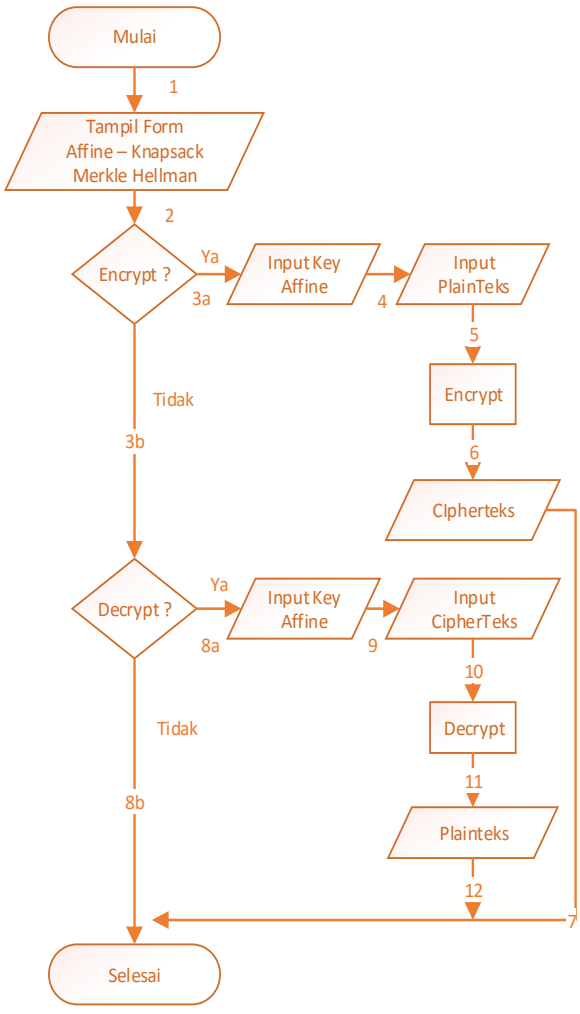
Gambar 5. Secure Your Teks (Decrypt)

Beberapa hasil pengujian terhadap aplikasi tersebut dapat dilihat pada Tabel 5. Tabel tersebut menggambarkan beberapa plainteks yang telah berhasil di enkripsi dengan aplikasi dan juga sukses kembali kebentuk semula dalam proses dekripsi. Penerapan kombinasi knapsack merkle hellman dan affine cipher berhasil melakukan enkripsi terhadap sebuah teks menjadi cipherteks yang sulit dimaknai dan juga berhasil untuk mengembalikan ke teks semula melalui proses dekripsi.

Tabel 5. Hasil pengujian aplikasi

Plainteks	Hasil Enkripsi (Cipherteks)	Hasil Dekripsi	Ket
Implementasi SUKSES	96 192 216 695 331 192 331 208 268 445 468 383 447 258 362 104 258 559 258	Implementasi SUKSES	✓
Jurnal JTIIK	304 572 260 208 445 695 447 304 346 96 96 104	Jurnal JTIIK	✓
Jurnal Teknologi Informasi dan Ilmu Komputer	304 572 260 208 445 695 447 346 331 391 208 200 695 200 339 383 447 96 208 323 200 260 192 445 468 383 447 315 445 208 447 96 695 192 572 447 104 200 192 216 572 268 331 260	Jurnal Teknologi Informasi dan Ilmu Komputer	✓
Universitas Brawijaya	362 208 383 146 331 260 468 383 268 445 468 447 227 260 445 354 383 591 445 378 445	Universitas Brawijaya	✓

Keterangan : ✓ = sesuai



Gambar 6. Flowchart Aplikasi Affine Knapsack

Secara keseluruhan, alur dalam menjalankan aplikasi kriptografi yang telah dibuat dapat dilihat pada Gambar 6, yang merupakan flowchart dari aplikasi tersebut. Dimana terdapat dua pilihan, yaitu apakah akan melakukan enkripsi data ataupun dekripsi data, yang mana tiap-tiap pilihan akan terdapat alur proses tersendiri seperti yang terlihat dalam Gambar 6 tersebut.

5. KESIMPULAN

Berdasarkan penjelasan pada bagian-bagian sebelumnya, khususnya pada bagian hasil dan pembahasan disimpulkan bahwa implementasi kombinasi affine cipher dan knapsack merkle hellman dapat dilakukan, salah satunya dibuktikan dengan dilakukan perhitungan secara manual dan dengan aplikasi yang telah dirancang. Proses enkripsi diawali dengan menggunakan metode affine cipher, yang kemudian cipherteks dari hasil metode tersebut dienkripsi lagi dengan knapsack merkle hellman sehingga menghasilkan cipherteks yang baru yang jauh berbeda dengan bentuk aslinya / plainteks. Selain itu, pada proses dekripsi juga berhasil mengembalikan cipherteks menjadi bentuk semula (plainteks) melalui tahapan dekripsi, yang diawali dengan dekripsi knapsack merkle hellman yang dilanjutkan dengan dekripsi affine cipher. Oleh karena itu, implementasi kombinasi affine dan knapsack merkle hellman tersebut dapat digunakan untuk mengamankan sebuah data.

6. DAFTAR PUSTAKA

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

ALI, F. M. 2014. Combination of Classical Cipher with Stream Cipher for Improving Data Security. *AL-Qadisiyha Journal For Science Vol.19 No. 3*, 190-197.

ARIYUS, D. 2008. *Pengantar Ilmu Kriptografi : Teori, Analisis dan Implementasi.* Yogyakarta: Andi.

BABU, S. A. 2017. Modification Affine Ciphers Algorithm For Cryptography Password. *International Journal of Research In Science & Engineering Volume: 3 Issue: 2*, 346-351.

FADLAN, M., & DEBY, K. 2014. Rekayasa Aplikasi Steganografi Untuk Teks Lagu Pada File Audio MP3 Dengan Menggunakan Metode Least Significant Bit. *Seminar Nasional Teknologi Informasi dan Multimedia* (hal. 1.14-29 – 1.14-32). Yogyakarta: STMIK AMIKOM.

HANDAYANI, E., PRATITIS, L. W., NUR, A., MASHURI, S. A., & NUGROHO, B. 2017. Perancangan Aplikasi Kriptografi Berbasis Web Dengan Algoritma Double Caesar Cipher Menggunakan Tabel Ascii. *Seminar Nasional Teknologi Informasi dan Multimedia* (hal. 241-246). Yogyakarta: STMIK Amikom.

HIDAYAT, A., AKMAL, & ROSYADI, R. 2016. Cryptography Asymmetries Merkle-Hellman Knapsack Digunakan untuk Enkripsi dan Dekripsi Teks. *Prosiding Seminar Nasional MIPA*, (hal. 66-69). Jatinangor.

MOKHTARI, M., & HASAN, N. 2012. Analysis and Design of Affine and Hill Cipher. *Journal of Mathematics Research, Vol. 4, No. 1*, 67-77.

MOON, S. K., & KAWITKAR, R. S. 2007. Data Security using Data Hiding. *International Conference on Computational Intelligence and Multimedia Applications*, (hal. 247-251). Sivakasi, Tamil Nadu, India.

MUNIR, R. 2006. *Kriptografi.* Bandung: Informatika.

MURDANI. 2017. Perancangan Aplikasi Keamanan Data Teks Menggunakan Algoritma Merkle Hellman Knapsack. *Jurnal Pelita Informatika, Volume 16, Nomor 3*, 284-302.

SADIKIN, R. 2012. *Kriptografi Untuk Keamanan Jaringan dan Implementasinya dalam Bahasa Java.* Yogyakarta: Andi.