Journal Global Tecnology Computer
Vol 2, No. 3, Agustus 2023, Hal 87 - 96
ISSN: **2809-6118** (Online - Elektronik)
Website https://ejurnal.seminar-id.com/index.php/jogtc

Aplikasi Pengamanan Data Arsip Menggunakan Metode Merkle Hellman

Ilham Akbar

Program Studi Teknik Informatika, Fakultas Ilmu Komputer & Teknologi Informasi, Universitas Budi Darma, Medan, Indonesia
Jl. Sisingamangaraja No.338, Siti Rejo I, Kec. Medan Kota, Kota Medan, Sumatera Utara, Indonesia
Email: akbarilham118@gmail.com

Abstrak− Kriptografi adalah ilmu sekaligus seni untuk menjaga keamanan data atau pesan dengan cara menyandikannya ke dalam bentuk yang tidak dapat dimengerti lagi maknanya salah satunya adalah dengan enkripsi dan dekripsi. Enkripsi adalah proses penyamaran suatu pesan atau sandi menjadi bentuk yang tidak mempunyai makna untuk menjaga kerahasiaan, keamanan, atau keotentikan suatu data atau pesan. Pada proses ini plainteks atau teks asli dikonversikan menjadi cipherteks (teks tersandi). *Merkle-Hellman* merupakan Kriptosistem yang dibuat oleh *Merkle* dan *Hellman* pada tahun 1980 Walaupun sistem ini, dan beberapa variannya, telah dipecahkan sekitar awal tahun 1980, tetapi masih layak untuk dipelajari dengan berbagai alasan. Masalah yang mendasari matematika adalah masalah penjumlahan himpunan bagian dimana sangat berhubungan dengan masalah dari operasi pencarian. Suatu masalah bisa dideskripsikan sebagai berikut. Jika setiap elemen dari himpunan S adalah suatu bilangan integer positif. Diberikan suatu himpunan bagian dari S, penjumlah dari elemen terdekat dari bagian himpunan bilangan menghasilkan bilangan integer yang berkoresponden dengan himpunan bagiannya. Hal inilah yang mendasari membuat aplikasi untuk menjaga keamanan data arsip.

Kata Kunci: Aplikasi, Kriptografi, Enkripsi, Dekripsi, Merkle Hellman.

Abstract− Cryptography is the science and art of keeping data or messages secure by encoding them in a form that cannot be understood anymore, one of which is encryption and decryption. Encryption is the process of disguising a message or code into a form that has no meaning to maintain the confidentiality, security or authenticity of a data or message. In this process plaintext or original text is converted into ciphertext (encoded text). Merkle-Hellman is a cryptosystem created by Merkle and Hellman in 1980. Although this system, and several variants of it, were solved around the early 1980s, it is still worth studying for a variety of reasons. The problem that underlies mathematics is the subset addition problem which is closely related to the problem of the search operation. A problem can be described as follows. If every element of the set S is a positive integer. Given a subset of S, the sum of the nearest elements of the subset of numbers yields the integer corresponding to its subset. This is what underlies making applications to maintain the security of archival data.

Keywords: Applications, Cryptography, Encryption, Decryption, Merkle Hellman.

1. PENDAHULUAN

Seiring dengan perkembangan teknologi computer menyebabkan terkaitnya satu komputer dengan komputer lainnya. Hal ini membuka banyak peluang dalam pengembangan aplikasi komputer tetapi juga membuat peluang adanya ancaman terhadap perubahan data dan pencurian data. Dalam hal ini, kearsipan mengubah sistem manual ke sistem komputerisasi dalam era teknologi dan informasi. Karena sering dilakukannya pengiriman atau pertukaran data, maka dilakukan proses penyandian atau enkripsi pada data yang akan dikirimkan. Sebelum dikirim ke penerima melalui media email, penerima harus memiliki aplikasi yang sama untuk mendeskripsi data tersebut agar dapat dibaca dan dimengerti kembali. Hal ini dilakukan untuk melindungi data atau informasi tersebut dari segala bentuk ancaman yang tidak diinginkan. Pada PT. Centrin Online Prima terdapat data arsip seperti data *customer* yang bertujuan sebagai informasi dan dokumentasi. Sebagai sumber informasi yang dapat membantu mengingatkan pegawai apabila ada perubahan data sewaktu waktu.

Informasi yang diarsipkan akan dapat terjaga kerahasiaannya maka perlu dilakukan pengamanan pada data arsip, salah satunya adalah dengan cara melakukan proses penyandian (enkripsi) terhadap informasi yang akan diarsipkan dalam bentuk berbagai file data yang sesuai dengan kebutuhan. Ada beberapa metode enkripsi yang telah dipublikasikan, salah satu nya adalah metode Merkle Hellman merupakan sistem kripto yang dibuat oleh *Merkle* dan *Hellman* pada tahun 1980. Walaupun sistem ini dan beberapa variannya telah dipecahkan sekitar awal tahun 1980, tetapi masih layak untuk dipelajari dengan berbagai alasan. Metode *Merkle Hellman* telah banyak digunakan untuk memodelkan masalah di industri seperti pada kriptografi kunci publik, metode ini menggunakan algoritma asimetris dan memiliki 2 kunci utama, yakni kunci public dan privat.

2. METODOLOGI PENELITIAN

2.1 Tahapan Penelitian

Tahapan penelitian yang digunakan dalam penulisan ini dilakukan dengan beberapa tahap, yaitu:

a. Studi Pustaka
   Pada tahap studi pustaka, kegiatan yang dilakukan adalah mencari dan mengumpulkan data atau teori pendukung terhadap metode yang digunakan dalam penunjang kegiatan proses penelitian.
b. Analisis
   Tahap analisis ini adalah proses hitung menghitung terhadap metode yang digunakan dengan contoh yang dapat di

gunakan. Dalam tahap ini masih bersifat eksperimen.

c. Desain Sistem
   Pada tahap desain sistem, dilakukan perancangan sistem yang dibuat. Desain sistem dibuat dengan sederhana dan mudah dimengerti oleh user yang awam sekalipun.
d. Pengujian Sistem
   Pada tahap pengujian sistem, sistem yang telah dibuat diuji dengan berbagai data yang berbeda untuk memastikan sistem berjalan sesuai dengan harapan dan analisis.
e. Implementasi Sistem
   Pada tahap ini sistem telah siap untuk diimplementasikan sesuai kebutuhan, dan aplikasi telah siap untuk diimplementasikan.

2.2 Arsip

Arsip adalah setiap catatan yang tertulis, tercetak atau ketikan dalam bentuk huruf, angka, atau gambaran yang mempunyai arti atau tujuan serta sebagai bahan komunikasi dan informasi yang terekam pada kertas (kartu, formulir, surat-surat), kertas film, media komputer (disket, harddisk, piringan) (Depkes, 1971:43). Dalam kegiatan praktis pengertian arsip dapat dirumuskan sebagai berikut [5]:

a. Naskah yang dibuat oleh lembaga dan badan pemerintah dalam bentuk apapun, baik dalam keadaan tunggal maupun berkelompok dalam rangka pelaksanaan kegiatan pemerintah.
b. Naskah yang dibuat dan diterima oleh badan-badan swasta atau perorangan dalam keadaan tunggal maupun berkelompok dalam rangka pelaksanaan kehidupan kebangsaan.

Arsip dalam istilah Bahasa Indonesia ada yang menyebutkan sebagai warkat, dan ada juga istilah lain yang menyebutnya sebagai record atau file seperti diutarakan oleh Atmosudirjo (1970), sebagai berikut [6]:

a. File berarti wadah, tempat, almari kabinet atau kumpulan tertatur (systematic). Bahan-bahan arsip dan file juga berarti setiap pengaturan, penyortiran, penerbitan yang sistematis dan berurutan atas barang-barang, orang-orang, personal, kertas-kertas tertulis, dokumen, dan sebagainya.
b. Record berarti setiap catatan yang dicatat untuk disimpan dan setiap bahan yang tertulis dapat dipergunakan sebagai bukti atas suatu peristiwa atau kejadian. Tidak hanya itu, plat atau piringan hitam, pita rekaman, suatu berita acara serta laporan resmi termasuk kepada pengertian record.

2.3 Kriptografi

Kriptografi (cryptography) merupakan ilmu dan seni untuk menjaga pesan agar aman. (Cryptography is the art and science of keeping messages secure). “Crypto” berarti “secret” (rahasia) dan “graphy” berarti “writing” (tulisan). Para pelaku atau praktisi kriptografi disebut cryptographers. Sebuah algoritma kriptografik (Cryptographic algorithm), disebut cipher, merupakan persamaan matematik yang digunakan untuk proses enkripsi dan deskripsi. Biasanya kedua persamaan matematik (untuk enkripsi dan dekripsi) tersebut memiliki hubungan matematis yang cukup erat. Kriptografi adalah ilmu yang mempelajari bagaimana membuat suatu pesan yang dikirim pengirim dapat disampaikan kepada penerima dengan aman (Schneier, 1996). Menurut Bruce Schneier dalam Applied Cryptography (John Wiley & Sons, 1996), ”Kriptografi adalah seni dan ilmu untuk menjaga agar pesan rahasia tetap aman [9].

Kriptografi merupakan salah satu cabang ilmu algoritma matematika”. Para penggemar kriptografi sering disebut cryptographer, sedangkan kebalikannya adalah crypt-analyst yang berusaha memecahkan sandi kriptografi. Proses yang dilakukan untuk mengamankan sebuah pesan (yang disebut plaintext) menjadi pesan yang tersembunyi (disebut ciphertext) adalah enkripsi (encryption). Ciphertext adalah pesan yang sudah tidak dapat dibaca dengan mudah. Menurut ISO 7498-2, terminologi yang lebih tepat digunakan adalah “encipher”. Proses sebaliknya, untuk mengubah ciphertext menjadi plaintext, disebut dekripsi (decryption). Menurut ISO 7498-2, terminologi yang lebih tepat untuk proses ini adalah “decipher”.

Cryptanalysis adalah seni dan ilmu untuk memecahkan ciphertext tanpa bantuan kunci. Cryptanalyst adalah pelaku atau praktisi yang menjalankan cryptanalysis. Cryptology merupakan gabungan dari Cryptography dan Cryptanalysis. Cryptographic System (Cryptosystem) Cryptographic System atau cryptosystem adalah suatu fasilitas untuk mengkonversikan plaintext ke ciphertext dan sebaliknya. Dalam sistem ini, seperangkat parameter yang menentukan transformasi pencipheran tertentu disebut suatu set kunci. Proses enkripsi dan dekripsi diatur oleh satu atau beberapa kunci. kriptografi. Secara umum, kunci- kunci yang digunakan oleh proses pengenkripsian dan pendekripsian tidak perlu identik, tergantung pada sistem yang digunakan [10].

2.4 Metode Merkle Hellman

Merkle-Hellman merupakan Kriptosistem yang dibuat oleh Merkle dan Hellman pada tahun 1980 Walaupun sistem ini, dan beberapa variannya, telah dipecahkan sekitar awal tahun 1980, tetapi masih layak untuk dipelajari dengan berbagai alasan. Masalah yang mendasari matematika adalah masalah penjumlahan himpunan bagian dimana sangat berhubungan dengan masalah dari operasi pencarian [13]. Suatu masalah bisa dideskripsikan sebagai berikut. Jika setiap elemen dari himpunan S adalah suatu bilangan integer positif. Diberikan suatu himpunan bagian dari S, penjumlah dari elemen terdekat dari bagian himpunan bilangan menghasilkan bilangan integer yang berkoresponden dengan himpunan bagiannya. Langkah-langkah pengamanan Merkle Hellman adalah sebagai berikut [14]:

Journal Global Tecnology Computer
Vol 2, No. 3, Agustus 2023, Hal 87 - 96
ISSN: **2809-6118** (Online - Elektronik)
Website https://ejurnal.seminar-id.com/index.php/jogtc

a. Proses Enkripsi

Langkah-langkah proses enkripsi sebagai berikut:

1. Membuat Private Key (S, A, dan P)

Nilai S, A, dan P adalah variable untuk private key. Angka-angka bilangan bulat yang disusun dengan algoritma superincreasing linear. S terdiri dari beberapa angka tergantung dari jumlah digit binner yang digunakan. A adalah nilai (angka) bebas yang harus lebih besar dari jumlah keseluruhan nilai S dengan maksimal nilai 999. Sedangkan P adalah nilai (angka) bebas yang dapat diambil mulai dari angka 1 sampai dengan nilai A.

Tabel 1. Private Key

S	{2, 4, 7, 14, 28, 112, 224, 407} = $\sum S$ = 798
A	989
P	578

2. Membuat Public Key

Public key digunakan untuk menghitung hasil chipper data. *Public key* memiliki karakter yang sama dengan *private key* S. Jika *private key* dilambangkan dengan S, maka *public key* dapat dilambangkan dengan T. Karena itu *public key* memiliki deretan angka sebagai kunci untuk mencari chipper. Perhitungan public key seperti tabel di bawah ini:

Tabel 2. *Public Key*

S	**T = (P * Si) mod A**	
2	578 * 2 mod 989	167
4	578 * 4 mod 989	334
7	578 * 7 mod 989	90
14	578 * 14 mod 989	180
28	578 * 28 mod 989	360
112	578 * 112 mod 989	451
224	578 * 224 mod 989	902
407	578 * 407 mod 989	853

Maka hasil dari proses pembuatan *public key* adalah T = {167, 334, 90, 180, 360, 451, 902, 853}.

a) Merubah Plainteks ke Binner 8 Digit

Pada proses ini data perlu dirubah menjadi bentuk binner karena perhitungan *Merkle Hellman* menggunakan teknik binary sebagai proses enkripsi dan dekripsinya. Untuk mengubah data ke binary 8 digit, maka sebelumnya data dirubah ke kode ASCII. Langkah selanjutnya adalah mengubah kode ASCII tersebut menjadi kode binary 8 digit seperti di bawah ini:

Tabel 3. Data *Binary*

Huruf	**ASCII**	**Binary (Z)**
2	050	00110010
0	048	00110000
1	049	00110001
3	051	00110011
0	048	00110000
2	050	00110010
0	048	00110000
2	050	00110010
3	051	00110011
1	049	00110001

b) Menjumlahkan (Perkalian Binner dengan *Public Key*)

Untuk proses perhitungan data *chippertext*, terlebih dahulu harus melakukan pembagian *plaintext* ke dalam blok-blok berdasarkan jumlah elemen T. Diketahui jumlah elemen T sebanyak 8 elemen. Selanjutnya, setiap blok akan dikaitkan dengan setiap elemen T, sehingga diperoleh *chippertext* sebagai berikut:

Tabel 4. Perhitungan Data *Ciphertext*

Binary	$\sum z$ * **T**	**Ciphertext**
00110010	(0*167)+(0*334)+(1*90)+(1*180)+(0*360)+(0*451)+(1*902)+(0*853)	1172
00110000	(0*167)+(0*334)+(1*90)+(1*180)+(0*360)+(0*451)+(0*902)+(0*853)	270
00110001	(0*167)+(0*334)+(1*90)+(1*180)+(0*360)+(0*451)+(*902)+(1*853)	1123
00110011	(0*167)+(0*334)+(1*90)+(1*180)+(0*360)+(0*451)+(1*902)+(1*853)	2025
00110000	(0*167)+(0*334)+(1*90)+(1*180)+(0*360)+(0*451)+(0*902)+(0*853)	270

00110010	(0*167)+(0*334)+(1*90)+(1*180)+(0*360)+(0*451)+(1*902)+(0*853)	1172
00110000	(0*167)+(0*334)+(1*90)+(1*180)+(0*360)+(0*451)+(0*902)+(0*853)	270
00110010	(0*167)+(0*334)+(1*90)+(1*180)+(0*360)+(0*451)+(1*902)+(0*853)	1172
00110000	(0*167)+(0*334)+(1*90)+(1*180)+(0*360)+(0*451)+(0*902)+(0*853)	2025
00110010	(0*167)+(0*334)+(1*90)+(1*180)+(0*360)+(0*451)+(1*902)+(0*853)	1123

Proses di atas menunjukkan bahwa proses enkripsi data sudah selesai dilakukan. Hal terakhir yang dilakukan adalah menyajikan data *chippertext* dengan menyimpan kembali ke dalam bentuk dokumen teks. Jadi hasil proses Enkripsi dari pesan 2013020231 adalah C{1172, 270, 1123, 2025, 270, 1172, 270, 1172, 2025, 1123}.

b. Proses Dekripsi

Langkah-langkah dalam proses dekripsi dengan menggunakan metode *Merkle Hellman* adalah sebagai berikut :

1. Data Chippertext (O)
   Dalam melakukan proses dekripsi, terlebih dahulu harus ada data yang lengkap dari proses enkripsi. Selain itu diperlukan juga *private key* sebagai kunci untuk proses dekripsi data. Kode *Chippertext* adalah sebagai berikut: C{1172, 270, 1123, 2025, 270, 1172, 270, 1172, 2025, 1123}
2. Modular *Invers*
   Proses untuk mencari nilai modulo invers dari (p-1) dengan menggunakan metode *extended eucledian*, yaitu (P * M mod A = 1). Dalam proses dekripsi ini akan digunakan nilai p-1 sebesar 77. Nilai 77 diperoleh dari hasil perhitungan menggunakan metode *extended eucledian*, seperti tabel di bawah ini:

Tabel 5. Modular *Invers*

M	(P*M) mod A	
1	578 * 1 mod 989	578
2	578 * 2 mod 989	167
3	578 * 3 mod 989	745
…	…	…
77	578 * 77 mod 989	1

3. *Chipper* Data Mod A
   Proses berikutnya adalah proses mod, yaitu untuk data *chippertext* dengan nilai *invers* yang diperoleh sebelumnya.

Tabel 6. *Chipper* Data Mod A

O	M	(O * M) Mod A	
1172	77	1172 * 77 mod 989	245
270	77	270 * 77 mod 989	21
1123	77	1123 * 77 mod 989	428
2025	77	2025 * 77 mod 989	652
270	77	270 * 77 mod 989	21
1172	77	1172 * 77 mod 989	245
270	77	270 * 77 mod 989	21
1172	77	1172 * 77 mod 989	245
2025	77	2025 * 77 mod 989	652
1123	77	1123 * 77 mod 989	428

4. Mengurangkan Data dengan Nilai S
   Proses Pengurangan data (K) dengan nilai-nilai pada elemen S. Pengurangan terus dilakukan dari elemen yang paling besar hingga yang paling kecil. Hasil akhir dari pengurangan haruslah bernilai 0. Hasil akhir dimana pengurangan tidak nol, maka proses dekripsi dinyatakan gagal. Penyebab kegagalan dapat terjadi apabila kunci S tidak dibuat dengan metode siperincreasing linier. S = {2, 4, 7, 14, 28, 112, 224, 407}, K = {245, 21, 428, 652, 21, 245, 21, 245, 652, 428}

Tabel 7. Pengurangan Data dengan Nilai S

2	4	7	14	28	112	224	407	S
							245-407	**K**
						245-224		
					21-112	= 21		
				21-28				
			21-14					
		7-7	=7					
	0-4	=0						
0-2								

Journal Global Tecnology Computer
Vol 2, No. 3, Agustus 2023, Hal 87 - 96
ISSN: **2809-6118** (Online - Elektronik)
Website https://ejurnal.seminar-id.com/index.php/jogtc

0	**0**	**1**	**1**	**0**	**0**	**1**	**0**

Proses perhitungan pada tabel di atas dimulai dari kanan ke kiri, kolom yang diberi tanda *false* berarti pada elemen S kolom tersebut data tidak dapat dikurangkan dan akan bernilai *false* atau 0. Sedangkan kolom yang berisi data *true*, berarti data dapat dikurangkan dan bernilai *true* atau 1. Apabila hasil data tersebut diambil keseluruhan maka akan menghasilkan nilai "00110010" yang apabila dikembalikan ke kode desimal menjadi "50" dan ke *char* menjadi "2".Proses berikutnya, nilai V1 sampai V10 akan dedekomposisi menggunakan setiap nilai pada S. Dekomposisi ini dilakukan dengan cara pengurangan terhadap nilai terbesar sampai terkecil dan menghasilkan nilai Vi=0.

V1 = 245 - 407 = 245(0) | 245 - 224 = 21(1) | 21 - 112 = 21(0) | 21 - 28 = 21(0) | 21 - 14 = 7(1) | 7 - 7 = 0(1) | 0 - 4 = 0(0) | 0 - 2 = 0(0)
Maka diperoleh hasil = 00110010
V2 = 21 - 407 = 21(0) | 21 - 224 = 21(0) | 21 - 112 = 21(0) | 21 - 28 = 21(0) | 21 - 14 = 7(1) | 7 - 7 = 0(1) | 0 - 4 = 0(0) | 0 - 2 = 0(0)
Maka diperoleh hasil = 00110000
V3 = 428 - 407 = 21(1) | 21 - 224 = 21(0) | 21 - 112 = 21(0) |21 - 28 = 21(0) | 21 - 14 = 7(1) | 7 - 7 = 0(1) | 0 - 4 = 0(0) | 0 - 2 = 0(0)
Maka diperoleh hasil = 00110001
V4 = 652 - 407 = 245(1) | 245 - 224 = 21(1) | 21 - 112 = 21(0) | 21 - 28 = 21(0) | 21 - 14 = 7(1) | 7 - 7 = 0(1) | 0 - 4 = 0(0) | 0 - 2 = 0(0)
Maka diperoleh hasil = 00110011
V5 = 21 - 407 = 21(0) | 21 - 224 = 21(0) | 21 - 112 = 21(0) | 21 - 28 = 21(0) | 21 - 14 = 7(1) | 7 - 7 = 0(1) | 0 - 4 = 0(0) | 0 - 2 = 0(0)
Maka diperoleh hasil = 00110000
V6 = 245 - 407 = 245(0) | 245 - 224 = 21(1) | 21 - 112 = 21(0) | 21 - 28 = 21(0) | 21 - 14 = 7(1) | 7 - 7 = 0(1) | 0 - 4 = 0(0) | 0 - 2 = 0(0)
Maka diperoleh hasil = 00110010
V7 = 21 - 407 = 21(0) | 21 - 224 = 21(0) | 21 - 112 = 21(0) | 21 - 28 = 21(0) | 21 - 14 = 7(1) | 7 - 7 = 0(1) | 0 - 4 = 0(0) | 0 - 2 = 0(0)
Maka diperoleh hasil = 00110000
V8 = 245 - 407 = 245(0) | 245 - 224 = 21(1) | 21 - 112 = 21(0) | 21 - 28 = 21(0) | 21 - 14 = 7(1) | 7 - 7 = 0(1) | 0 - 4 = 0(0) | 0 - 2 = 0(0)
Maka diperoleh hasil = 00110010
V9 = 652 - 407 = 245(1) | 245 - 224 = 21(1) | 21 - 112 = 21(0) | 21 - 28 = 21(0) | 21 - 14 = 7(1) | 7 - 7 = 0(1) | 0 - 4 = 0(0) | 0 - 2 = 0(0)
Maka diperoleh hasil = 00110011
V10 = 428 - 407 = 21(1) | 21 - 224 = 21(0) | 21 - 112 = 21(0) | 21 - 28 = 21(0) | 21 - 14 = 7(1) | 7 - 7 = 0(1) | 0 - 4 = 0(0) | 0 - 2 = 0(0)
Maka diperoleh hasil = 00110001
Z = {00110010, 00110000, 00110001, 00110011, 00110000, 00110010, 00110000, 00110010, 00110011, 00110001}

5. Mengembalikan ke Data Asli
   Mengembalikan ke data asli adalah tahapan terakhir untuk menkonversi ke proses dekripsi. Adapun kode *binary* disusun dan dikonversi ke kode desimal lalu ke kode char.
   C = C{1172, 270, 1123, 2025, 270, 1172, 270, 1172, 2025, 1123}
   Z = {2013020231}

3. HASIL DAN PEMBAHASAN

Kriptografi terbagi atas dua proses utama yaitu proses enkripsi dan dekripsi, masing-masing proses memiliki algoritma berbeda. Algoritma *Merkle Hellman* memiliki beberapa perhitungan yang berbeda pada setiap proses enkripsi dan dekripsi.

3.1 Implementasi Algoritma *Merkle Hellman*

Kriptografi terbagi atas dua proses utama yaitu proses enkripsi dan dekripsi, masing-masing proses memiliki algoritma berbeda. Algoritma *Merkle Hellman* memiliki beberapa perhitungan yang berbeda pada setiap proses enkripsi dan dekripsi.

3.1.1 Proses Enkripsi

Adapun langkah-langkah proses enkripsi dari suatu data dengan menggunakan metode *Merkle Hellman* adalah sebagai berikut:

a. Membuat *Private Key* (S, A dan P)

Journal Global Tecnology Computer
Vol 2, No. 3, Agustus 2023, Hal 87 - 96
ISSN: **2809-6118** (Online - Elektronik)
Website https://ejurnal.seminar-id.com/index.php/jogtc

Nilai S, A, dan P adalah variable untuk *private key*. Angka-angka bilangan bulat yang disusun dengan algoritma superincreasing linear. S terdiri dari beberapa angka tergantung dari jumlah digit binner yang digunakan. A adalah nilai (angka) bebas yang harus lebih besar dari jumlah keseluruhan nilai S dengan maksimal nilai 999. Sedangkan P adalah nilai (angka) bebas yang dapat diambil mulai dari angka 1 sampai dengan nilai A.

Tabel 8. *Private Key*

S	{2, 4, 7, 14, 28, 112, 224, 407} = $\sum S$ = 798
A	989
P	578

b. Membuat *Public Key*
Public key digunakan untuk menghitung hasil chipper data. *Public key* memiliki karakter yang sama dengan *private key* S. Jika *private key* dilambangkan dengan S, maka *public key* dapat dilambangkan dengan T. Karena itu *public key* memiliki deretan angka sebagai kunci untuk mencari chipper. Perhitungan *public key* seperti tabel di bawah ini:

Tabel 9. *Public Key*

S	**T = (P * Si) mod A**	
2	578 * 2 mod 989	167
4	578 * 4 mod 989	334
7	578 * 7 mod 989	90
14	578 * 14 mod 989	180
28	578 * 28 mod 989	360
112	578 * 112 mod 989	451
224	578 * 224 mod 989	902
407	578 * 407 mod 989	853

Maka hasil dari proses pembuatan *public key* adalah T = {167, 334, 90, 180, 360, 451, 902, 853}.

c. Merubah Plainteks ke Binner 8 Bit
Pada proses ini data perlu dirubah menjadi bentuk binner karena perhitungan *Merkle Hellman* menggunakan teknik *binary* sebagai proses enkripsi dan dekripsinya. Untuk mengubah data ke binary 8 bit, maka sebelumnya data dirubah ke kode ASCII. Langkah selanjutnya adalah mengubah kode ASCII tersebut menjadi kode binary 8 bit, yaitu:

Tabel 10. Data *Binary*

Huruf	**ASCII**	**Binary (Z)**
I	073	01001001
L	076	01001100
H	072	01001000
A	065	01000001
M	077	01001101
A	065	01000001
K	075	01001011
B	066	01000010
A	065	01000001
R	082	01010010

d. Menjumlahkan (Perkalian Binner dengan *Public Key*)
Untuk proses perhitungan data *chippertext*, terlebih dahuluharus melakukan pembagian *plaintext* ke dalam blok-blok berdasarkan jumlah elemen T. Diketahui jumlah elemen T sebanyak 8 elemen. Selanjutnya, setiap blok akan dikaitkan dengan setiap elemen T, sehingga diperoleh *chippertext* sebagai berikut:

Tabel 11. Perhitungan Data *Chippertext*

Binary	$\sum Z$ * **T**	**Ciphertext**
01001001	(0*167)+(1*334)+(0*90)+(0*180)+(1*360)+(0*45 1)+(0*902)+(1*853)	1547
01001100	(0*167)+(1*334)+(0*90)+(0*180)+(1*360)+(1*45 1)+(0*902)+(0*853)	1145
01001000	(0*167)+(1*334)+(0*90)+(0*180)+(1*360)+(0*45 1)+(0*902)+(0*853)	694
01000001	(0*167)+(1*334)+(0*90)+(0*180)+(0*360)+(0*45 1)+(0*902)+(1*853)	1187
01001101	(0*167)+(1*334)+(0*90)+(0*180)+(1*360)+(1*45 1)+(0*902)+(1*853)	1998
01000001	(0*167)+(1*334)+(0*90)+(0*180)+(0*360)+(0*45 1)+(0*902)+(1*853)	1187
01001011	(0*167)+(1*334)+(0*90)+(0*180)+(1*360)+(0*45 1)+(1*902)+(1*853)	2449
01000010	(0*167)+(1*334)+(0*90)+(0*180)+(0*360)+(0*45 1)+(1*902)+(0*853)	1236
01000001	(0*167)+(1*334)+(0*90)+(0*180)+(0*360)+(0*45 1)+(0*902)+(1*853)	1187

01010010	(0*167)+(1*334)+(0*90)+(1*180)+(0*360)+(0*45 1)+(1*902)+(0*853)	1416

Proses di atas menunjukkan bahwa proses enkripsi data sudah selesai dilakukan. Hal terakhir yang dilakukan adalah menyajikan data *chippertext* dengan menyimpan kembali ke dalam bentuk dokumen teks. Jadi hasil proses Enkripsi dari kata ILHAM AKBAR adalah O{1547, 1145, 694, 1187, 1998, 1187, 2449, 1236, 1187, 1416}.

3.1.2 Proses Dekripsi

Adapun langkah-langkah dalam proses dekripsi dari suatu proses enkripsi dengan menggunakan metode *Merkle Hellman* adalah sebagai berikut:

a. Data *Chippertext*
Dalam melakukan proses dekripsi, terlebih dahulu harus ada data yang lengkap dari proses enkripsi. Selain itu diperlukan juga *private key* sebagai kunci untuk proses dekripsi data. Adapun kode *Chippertext* (O) adalah sebagai berikut:
1547, 1145, 694, 1187, 1998, 1187, 2449, 1236, 1187, 1416

b. Modular *Invers*
Proses untuk mencari nilai modulo invers dari (p-1) dengan menggunakan metode *extended eucledian*, yaitu (P * M mod A = 1). Dalam proses dekripsi ini akan digunakan nilai p-1 sebesar 77. Nilai 77 diperoleh dari hasil perhitungan menggunakan metode *extended eucledian*, seperti tabel di bawah ini:

Tabel 12. Modular *Invers*

M	(P*M) mod A	
1	578 * 1 mod 989	578
2	578 * 2 mod 989	167
3	578 * 3 mod 989	745
…	…	…
77	578 * 77 mod 989	1

c. *Chipper* Data Mod A
Proses berikutnya adalah proses mod, yaitu untuk data *chippertext* dengan nilai *invers* yang diperoleh sebelumnya. Proses *chippertext* data mod a dapat dilihat pada table di bawah ini:

Tabel 13. *Chipper* Data Mod A

O	M	(O * M) Mod A	
1547	77	1547 * 77 mod 989	439
1145	77	1145 * 77 mod 989	144
694	77	694 * 77 mod 989	32
1187	77	1187 * 77 mod 989	411
1998	77	1998 * 77 mod 989	551
1187	77	1187 * 77 mod 989	411
2449	77	2449 * 77 mod 989	663
1236	77	1236 * 77 mod 989	228
1187	77	1187 * 77 mod 989	411
1416	77	1416 * 77 mod 989	242

d. Mengurangkan Data Dengan Nilai S
Proses pengurangan data (K) dengan nilai-nilai pada elemen S. Pengurangan terus dilakukan dari elemen yang paling besar hingga yang paling kecil. Hasil akhir dari pengurangan haruslah bernilai 0. Hasil akhir dimana pengurangan tidak nol, maka proses dekripsi dinyatakan gagal. Penyebab kegagalan dapat terjadi apabila kunci S tidak dibuat dengan metode *superincreasing linier*. S = {2, 4, 7, 14, 28, 112, 224, 407}, K = {439, 144, 32, 411, 551, 411, 663, 228, 411, 242}

Tabel 14. Proses Pengurangan Data dengan Nilai S

2	**4**	**7**	**14**	**28**	**112**	**224**	**407**	**S**
							439-407	**K**
						32-224	=32	
					32-112			
				32-28				
			4-14	=4				
		4-7						
	4-4							
0-2	=0							
0	**1**	**0**	**0**	**1**	**0**	**0**	**1**	

Proses perhitungan pada tabel di atas dimulai dari kanan ke kiri, kolom yang diberi tanda *false* berarti pada elemen S kolom tersebut data tidak dapat dikurangkan dan akan bernilai *false* atau 0. Sedangkan kolom yang berisi data *true*, berarti data dapat dikurangkan dan bernilai *true* atau 1. Apabila hasil data tersebut diambil keseluruhan maka akan menghasilkan nilai "01001001" yang apabila dikembalikan ke kode desimal menjadi "84" dan ke *char* menjadi "I". Proses berikutnya, nilai V1 sampai V10 akan dekomposisi menggunakan setiap nilai pada S. Dekomposisi ini dilakukan dengan cara pengurangan terhadap nilai terbesar sampai terkecil dan menghasilkan nilai Vi=0.

V1 = 439– 407 (1)
= 32 – 224 (0)
= 32 – 112 (0)
= 32 – 28 (1)
= 4 – 14 (0)
= 4– 7 (0)
= 4 – 4 (1)
= 0 – 2 (0)
Maka diperoleh hasil = 01001001 = 73 = I
V2 = 144 – 407 (0)
= 144 – 224 (0)
= 144 – 112 (1)
= 32 – 28 (1)
= 4 – 14 (0)
= 4 – 7 (0)
= 4 – 4 (1)
= 0 – 2 (0)
Maka diperoleh hasil = 01001100 = 76 = L
V3 = 32 – 407 (0)
= 32 – 224 (0)
= 32 – 112 (0)
= 32 – 28 (1)
= 4 – 14 (0)
= 4 – 7 (0)
= 4 – 4 (1)
= 0 – 2 (0)
Maka diperoleh hasil = 01001000 = 72 = H
V4 = 411 – 407 (1)
= 4 – 224 (0)
= 4 – 112 (0)
= 4 – 28 (0)
= 4 – 14 (0)
= 4 – 7 (0)
= 4 – 4 (1)
= 0 – 2 (0)
Maka diperoleh hasil = 01000001 = 65 = A
V5 = 551 – 407 (1)
= 144 – 224 (0)
= 144 – 112 (1)
= 32 – 28 (1)
= 4 – 14 (0)
= 4 – 7 (0)
= 4 – 4 (1)
= 0 – 2 (0)
Maka diperoleh hasil = 01001101 = 77 = M
V6 = 411 – 407 (1)
= 4 – 224 (0)
= 4 – 112 (0)
= 4 – 28 (0)
= 4 – 14 (0)
= 4 – 7 (0)
= 4 – 4 (1)
= 0 – 2 (0)
Maka diperoleh hasil = 01000001 = 65 = A
V7 = 663 – 407 (1)
= 256 – 224 (1)

= 32 – 112 (0)
= 32 – 28 (1)
= 4 – 14 (0)
= 4 – 7 (0)
= 4 – 4 (1)
= 0 – 2 (0)

Maka diperoleh hasil = 01001011 = 75 = K

V8 = 228 – 407 (0)
= 228 – 224 (1)
= 4 – 112 (0)
= 4 – 28 (0)
= 4 – 14 (0)
= 4 – 7 (0)
= 4 – 4 (1)
= 0 – 2 (0)

Maka diperoleh hasil = 01000010 = 66 = B

V9 = 411 – 407 (1)
= 4 – 224 (0)
= 4 – 112 (0)
= 4 – 28 (0)
= 4 – 14 (0)
= 4 – 7 (0)
= 4 – 4 (1)
= 0 – 2 (0)

Maka diperoleh hasil = 01000001 = 65 = A

V10 = 242 – 407 (0)
= 242 – 224 (1)
= 18 – 112 (0)
= 18 – 28 (0)
= 18 – 14 (1)
= 4 – 7 (0)
= 4 – 4 (1)
= 0 – 2 (0)

Maka diperoleh hasil = 01010010 = 82 = R

Z = {01001001, 01001100, 01001000, 01000001, 01001101, 01000001, 01001011, 01000010, 01000001, 01010010}

e. Mengembalikan ke Data Asli

Mengembalikan ke data asli adalah tahapan terakhir untuk menkonversi enkripsi ke proses dekripsi. Adapun kode *binary* disusun dan dikonversi ke kode desimal lalu ke kode *char*.

O = {1547, 1145, 694, 1187, 1998, 1187, 2449, 1236, 1187, 1416}

Z = {ILHAM AKBAR}

4. KESIMPULAN

Adapun kesimpulan yang dapat diambil dari penelitian yang telah dilakukan yaitu, Penerapan kriptografi Merkle Hellman dalam mengamankan data arsip untuk mengubah isi pesan menjadi bentuk simbol-simbol yang tidak dapat dimengerti maknanya dengan menggunakan operasi *enkripsi* dan *deskripsi* antara *plainteks* dengan kunci yang dimasukkan, hasil dari enkripsi ini akan disisipkan atau disembunyikan sehingga keamanan data arsip lebih terjaga kerahasiaan nya. Metode Merkle Hellman menggunakan panjang kunci dari 1 sampai 256 *byte* yang digunakan untuk menginisialisasi tabel dengan panjang 256 *byte*.

REFERENCES

[1] H. Jogiyanto, *Pengertian Aplikasi*. Yogyakarta: Andi, 1999.
[2] K. B. B. Indonesia., *Pengolahan Data*. jakarta: Pustaka Amani, 1998.
[3] J. Sasongko, “Pengamanan Data Informasi,” vol. 10, p. 3, 2005.
[4] BKPSDMD, “Keamanan Data dan Informasi,” 2017. bkpsdmd.babelprov.go.id/content/keamanan-data-informasi.
[5] D. K. RI, *Bentuk Pokok Penyelenggaraan Sistem Kesehatan Nasional*. Jakarta, 1971.
[6] A. S. Prayudi, *Dasar-Dasar Ilmu Kearsipan*. jakarta: Gunung Agung, 1970.
[7] D. News, “Pengertian, Jenis, Tujuan Arsip dan Pengertian Kearsipan,” 2020. https://www.diwarta.com/2020/08/30/pengertian-jenis-tujuan-arsip-dan-pengertian- kearsipan.html.
[8] Basipda, “Jenis-Jenis Arsip,” 2020. https://basipda.bekasikab.go.id/berita-jenisjenis- arsip.html.

[9] S. Bruce, *Applied Cryptography*. 1996.
[10] "Fasilkom," vol. 4, pp. 49–50, 2006.
[11] Dafid, "Kriptografi Kunci Simetris Dengan Menggunakan Algoritma Crypton," *STMIK GI MDP*, vol. 2, p. 3, 2006.
[12] "Teknik Dasar Kriptografi," *docplayer*, 2020. https://docplayer.info/46059854-Teknik- dasar-kriptografi-algoritma-kriptografi-modern-bagian-1-subsitusi-tabel-substitusi- substitusi-blocking-permutasi-ekspansi-pemampatan.html.
[13] A. Hidayat, "Kriptosistem Knapsack," 2007.
[14] H. Mukhtar, *Kriptografi Untuk Keamanan Data*. Yogyakarta: Deepublish, 2018.
[15] D. Nofriansyah, "Application to determination of scholarship Merkle Hellman Method," *Int. J. Artif. Intell. Res.*, 2017.
[16] Sulindawati & Fathoni, "Pengantar Analisa Perancangan Sistem," *SAINTIKOM*, vol. 9, 2010.
[17] Rossa A.S M.Shalahuddin, *Rekayasa Perangkat Lunak Berorientasi Objek*. Bandung, 2014.