Jurnal Cyber Tech
Vol.x, No.x, September 2019, pp. xx~xx
P-ISSN :
E-ISSN :

Penerapan Metode Merkle Hellman Untuk Pengamanan Data Nilai Siswa Pada SMKS Teladan Sumatera Utara 2

Imorananta Berutu *, Badrul Anwar **, Tugiono **
* Program StudiSistemInformasi, STMIK Triguna Dharma
** Program StudiSistemInformasi, STMIK Triguna Dharma

Article Info
Article History:
Received
Revised
Accepted

Keyword:
Kriptografi, Merkle Hellman, Data Nilai Siswa, Desktop

ABSTRACT

Nilai merupakan suatu bentuk penghargaan serta keadaan yang bermanfaat bagi siswa sebagai penentu dan acuan dalam melakukan suatu tindakan. Melalui penilaian, siswa dapat mengetahui sejauh mana telah berhasil mengikuti pelajaran yang diberikan oleh guru.Apakah siswa merasa puas atas hasil yang diperolehnya. Dan disini nilai dapat dimanipulasi atau diubah orang yang tidak bertanggung jawab.

Dan untuk mengatasi permasalahan tersebut perlu adanya pengamanan data nilai agar data tersebut tetap aman dan tidak dimanipulasi atau diubah oleh siapa pun yaitu dengan menggunakan dengan menggunakan kriptografi. Kriptografi merupakan seni dan ilmu untuk menjaga keamanan data dengan mengubahnya menjadi kode tertentu. Dan untuk penerapan kriptografi terdapat beberapa metode atau algoritma, salah satunya adalah Merkle Hellman.

Dengan pembuatan Kriptografi, maka nilai yang dalam bentuk excel dapat diamankan menjadi kode tertentu dan tidak akan dapat diakses oleh siapapun kecuali yang bersangkutan.

Corresponding Author :
Nama : Imorananta Berutu
Kantor : STMIK Triguna Dharma
Program Studi : SistemInformasi
E-Mail : ranantabrutu@gmail.com

1. PENDAHULUAN

Nilai merupakan suatu bentuk penghargaan serta keadaan yang bermanfaat bagi siswa sebagai penentu dan acuan dalam melakukan suatu tindakan. Melalui penilaian, siswa dapat mengetahui sejauh mana telah berhasil mengikuti pelajaran yang diberikan oleh guru.Apakah siswa merasa puas atas hasil yang diperolehnya. Bila hasilnya memuaskan akan menyenangkan dan dapat memotivasi siswa untuk belajar lebih giat lagi sementara bila hasil tidak memuaskan maka ia akan berusaha agar penilaian berikutnya memperoleh hasil yang memuaskan. Dan guru harus mampu menjaga data nilai agar tetap aman dan tidak dapat diubah oleh siapapun. Ada cara untuk mengamankan data nilai agar data tersebut tetap aman dan tidak diubah oleh pihak lain, yaitu dengan menggunakan Kriptografi.[2]

Kriptografi merupakan seni dan ilmu untuk menjaga keamanan data dengan mengubahnya menjadi kode tertentu, dan hanya ditujukan untuk orang yang hanya memiliki sebuah kunci untuk mengubah kode itu kembali, yang berfungsi dalam menjaga kerahasiaan data atau pesan.[3] Dan untuk penerapan kriptografi terdapat beberapa metode atau algoritma, salah satunya adalah Merkle Hellman. Merkle Hellman adalah salah satu sistem kriptografi yang menggunakan tipe kunci asimetris. Pada sistem Merkle Hellman ini, kunci yang digunakan adalah 2 kunci yang berbeda. Satu kunci untuk mengenkripsi dan satu kunci untuk mendekripsi.[4]

2 KAJIAN PUSTAKA

2.1 Pengertian Kriptografi

Kriptografi *(cryptography)* berasal dari bahasa Yunani yaitu *cryptos* yang artinya "*secret*"(yang tersembunyi) dan *graphein* yang artinya "*writting*"*(*tulisan). Jadi Kriptografi berarti "secret writting"(tulisan rahasia).[5] Kriptografi adalah sebuah teknik rahasia dalam penulisan, dengan menggunakan karakter khusus, dan menggunakan huruf dan karakter di luar bentuk aslinya ataupun dengan metode-metode yang lain yang hanya bisa dipahami pihak-pihak yang memproses kunci. Kriptografi merupakan sebuah studi teknik matematika yang berkaitan dengan aspek

keamanan informasi seperti kerahasiaan, otentikasi entitas serta otentikasi keaslian data dan integritas data. Kriptografi tidak hanya penyediaan keamanan informasi saja, tetapi juga sebuah himpunan teknik-teknik.[6]

2.2 *Merkle Hellman*

Merkle-Hellman Knapsack merupakan kripto sistem yang menggunakan algoritma asimetris dan memiliki 2 kunci utama, yakni kunci publik dan kunci privat. 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). Kelebihan lain adalah pada efisiensi jumlah kunci publik. Jika terdapat n user, maka hanya membutuhkan 1 (satu) kunci publik, sehingga untuk jumlah user yang sangat banyak, sistem ini sangat efisien.

2.2.1 Proses Enkripsi

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

1. Membuat *private Key* (S, A, P)
   Nilai S, A, P adalah bilangan bulat yang disusun dengan algoritma *superincreasing linear*. S terdiri dari beberapa angka tergantung dari jumlah digit biner yang digunakan. A adalah nilai (angka) bebas yang harus lebih besar dari jumlah keseluruhan nilai S. Sedangkan P adalah nilai (angka) bebas yang dapat diambil mulai dari angka 1 sampai nilai A.
   Membuat urutan s =($^{s}1$, $^{s}2$,....,^{s}n)
   a > $\sum^{n}$**i=1 Si**……2.2

2. Membuat *Public Key*
   Public Key digunakan untuk menghitung hasil *chipper* data. *Public Key* memiliki karakter yang sama dengan *private key*. 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*.
   T* = P * Si *mod A.... 2.3

3. Merubah Plainteks ke Biner 8 Digit
   Pada proses ini data perlu diubah menjadi bentuk biner karena perhitungan Merkle Hellman menggunakan tehnik *binary* sebagai proses enkripsi dan dekripsinya. Untuk mengubah data ke *binery* 8 digit, maka sebelumnya data dirubah ke kode ASCII.

4. Menjumlahkan (Perkalian 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.*
   C = $\sum^{z}$ * **T**….2.4

2.2.2 Proses Enkripsi

Adapun 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.

2. Modular *Invers* (M)
   Proses untuk mencari nilai *modulo invers* dari (p^{-1}) dengan menggunakan metode *extended eucledian*. Dalam proses dekripsi ini akan digunakan nilai p^{-1}. Nilai M diperoleh dari hasil perhitungan menggunakan metode *extended eucledian.*
   M = (P * M mod A = 1)……..2.1

3. *Chipper* Data Mod Q
   Proses berikutnya adalah proses mod, yaitu untuk data *chippertext* dengan nilai *invers* yang diperoleh sebelumnya.
   K = (O * M) mod A....2.2

4. Mengurangkan Data Dengan Nilai S

Proses pengurangan data (K) dengan nilai-nilai pada elemen S. Pengurangan terus dilakukan darri elemen yang paling besar hingga yang paling kecil. Hasil akhir dari pengurangan haruslah 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*.

3. METODOLOGI PENELITIAN

3.1 Metode Penelitian

Berikut metode penelitian yang digunakan dalam penelitian ini adalah:

1. Studi Kepustakaan (*Library Research*)
   Studi Kepustakaan merupakan salah satu elemen yang mendukung sebagai landasan teoritis peneliti untuk mengkaji masalah yang dibahas. Dalam hal ini, peneliti menggunakan beberapa sumber kepustakaan diantaranya: Buku, Jurnal, dan sumber lainnya.

2. Observasi
   Observasi merupakan teknik pengumpulan data dengan melakukan tinjauan langsung ketempat studi kasus dimana akan dilakukan penelitian. Dalam hal ini peneliti melakukan observasi di SMKS Teladan Sumatera Utara 2.[20]

3. Wawancara
   Teknik wawancara ini dilakukan untuk mendapatkan informasi tambahan dari pihak-pihak yang memiliki wewenang dan berinteraksi langsung dengan sistem yang akan dirancang sebagai sumber data.

3.3.2 Proses Enkripsi

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

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

Tabel 3.2 *Private Key*

S	{ 3, 5, 8, 16, 29, 53, 114, 224 }=Σ^{s} = 452
A	621
P	241

2. Membuat *Public Key*

Tabel 3.2 *Public Key*

S	T= (P * Si) mod A	
3	241 * 3 mod 621	102
5	241 * 5 mod 621	584
8	241 * 8 mod 621	65
16	241 * 16 mod 621	130
29	241 * 29 mod 621	158
53	241 * 53 mod 621	353
114	241 * 114 mod 621	150
224	241 * 224 mod 621	578

Maka Hasil proses *Public Key* adalah: T { 102, 584, 65, 130, 158, 353, 150, 578 }

3. Merubah *Plaintext* ke Binner 8 Digit

Plaintext: AJI SYAIBAN TKJ 90

Tabel 3.2 Data *Binnary*

Plaintext	**ASCII**	**Binnary**
A	65	01000001
J	74	01001010
I	73	01001001
SPACE	32	00100000
S	83	01010011
Y	89	01011001
A	65	01000001
I	73	01001001
B	66	01000010
A	65	01000001
N	78	01001110
SPACE	32	00100000
T	84	01010100
K	75	01001011
J	74	01001010
SPACE	32	00100000
9	9	00001010
0	0	00000000

4. Menjumlahkan (Perkalian Biner dengan *Public Key*)

Tabel 3.2 Proses Perhitungan Data *Chippertext*

Binary (z)	$\sum z_* T$	**Chippertext**
01000001	(0 * 102) + (1* 584) + (0* 65) + (0* 130) + (0* 158) + (0* 353) + (0* 150) + (1* 578)	1162
01001010	(0 * 102) + (1* 584) + (0* 65) + (0* 130) + (1* 158) + (0* 353) + (1* 150) + (0* 578)	892
01001001	(0 * 102) + (1* 584) + (0* 65) + (0* 130) + (1* 158) + (0* 353) + (0* 150) + (1* 578)	1320
00100000	(0 * 102) + (0* 584) + (1* 65) + (0* 130) + (0* 158) + (0* 353) + (0* 150) + (0* 578)	65
01010011	(0 * 102) + (1* 584) + (0* 65) + (1* 130) + (0* 158) + (0* 353) + (1* 150) + (1* 578)	1442
01011001	(0 * 102) + (1* 584) + (0* 65) + (1* 130) + (1* 158) + (0* 353) + (0* 150) + (1* 578)	1450
01000001	(0 * 102) + (1* 584) + (0* 65) + (0* 130) + (0* 158) + (0* 353) + (0* 150) + (1* 578)	1162
01001001	(0 * 102) + (1* 584) + (0* 65) + (0* 130) + (1* 158) + (0* 353) + (0* 150) + (1* 578)	1320

Tabel 3.2 Proses Perhitungan Data *Chippertext* (Lanjutan)

Binary (z)	$\sum z_* T$	**Chippertext**
01000010	(0 * 102) + (1* 584) + (0* 65) + (0* 130) + (0* 158) + (0* 353) + (1* 150) + (0* 578)	734
01000001	(0 * 102) + (1* 584) + (0* 65) + (0* 130) + (0* 158) + (0* 353) + (0* 150) + (1* 578)	1162
01001110	(0 * 102) + (1* 584) + (0* 65) + (0* 130) + (1* 158) + (1* 353) + (1* 150) + (0* 578)	1245
00100000	(0 * 102) + (0* 584) + (1* 65) + (0* 130) + (0* 158) + (0* 353) + (0* 150) + (0* 578)	65
01010100	(0 * 102) + (1* 584) + (0* 65) + (1* 130) + (0* 158) + (1* 353) + (0* 150) + (0* 578)	1067
01001011	(0 * 102) + (1* 584) + (0* 65) + (0* 130) + (1* 158) + (0* 353) + (1* 150) + (1* 578)	1470
01001010	(0 * 102) + (1* 584) + (0* 65) + (0* 130) + (1* 158) + (0* 353) + (1* 150) + (0* 578)	892
00100000	(0 * 102) + (0* 584) + (1* 65) + (0* 130) + (0* 158) + (0* 353) + (0* 150) + (0* 578)	65
00001010	(0 * 102) + (0* 584) + (0* 65) + (0* 130) + (1* 158) + (0* 353) + (1* 150) + (0* 578)	308
00000000	(0 * 102) + (0* 584) + (0* 65) + (0* 130) + (0* 158) + (0* 353) + (0* 150) + (0* 578)	0

C = {1162, 892, 1320, 65, 1442, 1450, 1162, 1320, 734, 1162, 1245, 65, 1067, 1470, 892, 65, 308, 0}

3.3.3 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 *private key* juga dibutuhkan sebagai kunci untuk proses dekripsi data.
   Kode *chippertext* adalah sebagai berikut:

C = {1162, 892, 1320, 65, 1442, 1450, 1162, 1320, 734, 1162, 1245, 65, 1067, 1470, 892, 65, 308, 0}

2. Modular Invers(M)
   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}). Nilai (p^{-1}) diperoleh dari hasil perhitungan menggunakan metode *extended eucledian*, seperti table dibawah ini:

Tabel 3.3 Proses Perhitungan M Invers

M	**(P * M) mod A**	
1	241 * 1 mod 621	241
2	241 * 2 mod 621	482
3	241 * 3 mod 621	102
…	…..	…
67	241 * 67 mod 621	1

3. Chipper Data Mod Q

Proses berikutnya adalah proses mod, yaitu data *chippertext* dengan nilai invers yang diperoleh sebelumnya.

Tabel 3.3 Chipper Data Mod Q

Chipper (O)	**M**	**K = (O * M) mod A**	
1162	67	1162 * 67 mod 621	229
892	67	892 * 67 mod 621	148
1320	67	1320 * 67 mod 621	258
65	67	65 * 67 mod 621	8
1442	67	1442 * 67 mod 621	359
1450	67	1450* 67 mod 621	274
1162	67	1162* 67 mod 621	229
1320	67	1320 * 67 mod 621	258
734	67	734 * 67 mod 621	119
1162	67	1162 * 67 mod 621	229
1245	67	1245 * 67 mod 621	201
65	67	65 * 67 mod 621	8
1067	67	1067 * 67 mod 621	74
1470	67	1470 * 67 mod 621	372
892	67	892 * 67 mod 621	148
65	67	65* 67 mod 621	8
308	67	308* 67 mod 621	143
0	67	0 * 67 mod 621	0

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 *superincreasing linier.*

S = {3, 5, 8, 16, 29, 53, 114, 224}

K = {229, 148, 258, 8, 359, 274, 229, 258, 119, 229, 201, 8, 74, 372, 148, 8, 143, 0}

Tabel 3.3 Proses Pengurangan *Chippertext*

3	**5**	**8**	**16**	**29**	**53**	**114**	**224**	**S**
							229-224	K
						5-114	=5	
					5-53			
				5-29				
			5-16					
		5-8						
	5-5							
0-3	=0							
0	1	0	0	0	0	0	1	

Proses perhitungan pada table diatas dimulai dari kolom kanan lalu kekolom kiri, kolom K dikurangi dengan kolom S, jika kolom K dan kolom S dapat dikurangkan dan menghasilkan nilai positif maka akan menghasilkan bilangan binernya adalah *true* atau 1, jika kolom K dan kolom S saat dikurangkan mendapatkan hasil negative maka bilangan binernya adalah *false* atau 0, pada kolom selanjutnya hasil dari pengurangan kolom sebelumnya akan dikurangkan dengan bilangan S, lalu teruskan pengurangan pada setiap kolom. Apabila hasil data tersebut diambil keseluruhan maka akan menghasilkan nilai "01000001" yang apabila dikembangkan ke kode decimal menjadi "65" dan ke char menjadi "S". Proses berikutnya, nilai v1 sampai v18 akan dikomposisi menggunakan setiap nilai pada S. Dekomposisi ini dilakukan dengan cara pengurangan terhadap nilai tersebut sampai terkecil dan menghasilkan vi=0.

V1 = 229 – 224 (1)
= 5 – 114 (1)
= 5 – 53 (0)
= 5 – 29 (0)
= 5 – 16 (0)
= 5 – 8 (0)

= 5 – 5 (1)
= 0 – 3 (0)

Maka diperoleh hasil = 01000001 = 65 = A

V2 = 148 – 224 (0)
= 148 – 114 (1)
= 34 – 53 (0)
= 34 – 29 (1)
= 5 – 16 (0)
= 5 – 8 (0)
= 5 – 5 (1)
= 0 – 3 (0)

Maka diperoleh hasil = 01001010 = 74 = J

V3 = 258 – 224 (1)
= 34 – 114 (1)
= 34 – 53 (0)
= 34 – 29 (1)
= 5 – 16 (0)
= 5 – 8 (0)
= 5 – 5 (1)
= 0 – 3 (0)

Maka diperoleh hasil = 01001001 = 73 = I

V4 = 8 – 224 (0)
= 8 – 114 (0)
= 8 – 53 (0)
= 8 – 29 (0)
= 8 – 16 (0)
= 8 – 8 (1)
= 0 – 5 (0)
= 0 – 3 (0)

Maka diperoleh hasil = 00100000 = 32 = SPACE

V5 = 359 – 224 (1)
= 135 – 114 (1)
= 21 – 53 (0)
= 21 – 29 (0)
= 21 – 16 (1)
= 5 – 8 (0)
= 5 – 5 (1)
= 0 – 3 (0)

Maka diperoleh hasil = 01010011 = 83= S

V6 = 274 – 224 (1)
= 50 – 114 (0)
= 50 – 53 (0)
= 50 – 29 (1)
= 21 – 16 (1)
= 5 – 8 (0)
= 5 – 5 (1)
= 0 – 3 (0)

Maka diperoleh hasil = 01011001 = 89 = Y

V7 = 229 – 224 (1)
= 5 – 114 (0)
= 5 – 53 (0)
= 5 – 29 (0)
= 5 – 16 (0)
= 5 – 8 (0)
= 5 – 5 (1)
= 0 – 3 (0)

Maka diperoleh hasil = 01000001= 65 = A

V8 = 258 – 224 (1)
= 34 – 114 (0)
= 34 – 53 (0)
= 34 – 29 (1)

= 5 – 16 (0)
= 5 – 8 (0)
= 5 – 5 (1)
= 0 – 3 (0)

Maka diperoleh hasil = 01001001 = 73 = I

V9 = 119 – 224 (0)
= 119 – 114 (1)
= 5 – 53 (0)
= 5 – 29 (0)
= 5 – 16 (0)
= 5 – 8 (0)
= 5 – 5 (1)
= 0 – 3 (0)

Maka diperoleh hasil = 01000010 = 66 = B

V10 = 229 – 224 (1)
= 5 – 114 (0)
= 5 – 53 (1)
= 5 – 29 (0)
= 5 – 16 (0)
= 5 – 8 (0)
= 5 – 5 (1)
= 0 – 3 (0)

Maka diperoleh hasil = 01000001 = 65 =A

V11 = 201 – 224 (0)
= 201 – 114 (1)
= 87 – 53 (1)
= 34 – 29 (1)
= 5 – 16 (0)
= 5 – 8 (0)
= 5 – 5 (1)
= 0 – 3 (0)

Maka diperoleh hasil = 01001110 = 78 = N

V12= 8 – 224 (0)
= 8 – 114 (0)
= 8 – 53 (0)
= 8 – 29 (0)
= 8 – 16 (0)
= 8 – 8 (1)
= 0 – 5 (0)
= 0 – 3 (0)

Maka diperoleh hasil = 00100000 = 32 = SPACE

V13= 74 – 224 (0)
= 74 – 114 (0)
= 74 – 53 (1)
= 21– 29 (0)
= 21 – 16 (1)
= 5 – 8 (0)
= 5 – 5 (1)
= 0 – 3 (0)

Maka diperoleh hasil = 01010100 = 84 = T

V14= 372 – 224 (1)
= 148 – 114 (1)
= 34 – 53 (0)
= 34– 29 (1)
= 5 – 16 (0)
= 5 – 8 (0)
= 5 – 5 (1)
= 0 – 3 (0)

Maka diperoleh hasil = 01001011 = 75 = K

V15 = 148 – 224 (0)
= 148 – 114 (1)

= 34 – 53 (0)
= 34– 29 (1)
= 5 – 16 (0)
= 5 – 8 (0)
= 5 – 5 (1)
= 0 – 3 (0)

Maka diperoleh hasil = 01001010 = 74 = J

V16= 8 – 224 (0)
= 8 – 114 (0)
= 8 – 53 (0)
= 8 – 29 (0)
= 8 – 16 (0)
= 8 – 8 (1)
= 0 – 5 (0)
= 0 – 3 (0)

Maka diperoleh hasil = 00100000 = 32 = SPACE

V17= 143 – 224 (0)
= 143 – 114 (1)
= 29 – 53 (0)
= 29 – 29 (1)
= 0 – 16 (0)
= 0 – 8 (0)
= 0 – 5 (0)
= 0 – 3 (0)

Maka diperoleh hasil = 00001010 = 9 = 9

V18 = 0 – 224 (0)
= 0 – 114 (0)
= 0 – 53 (0)
= 0 – 29 (0)
= 0 – 16 (0)
= 0 – 8 (0)
= 0 – 5 (0)
= 0 – 3 (0)

Maka diperoleh hasil = 00000000 = 0 = 0

5. Mengembalikan ke Data Asli
   Mengembalikan ke data asli adalah hubungan tahapan terakhir untuk menkonversi enkripsi ke proses dekripsi. Adapun kode *binary* disusun dan dikonversikan ke kode decimal lalu ke kode char.

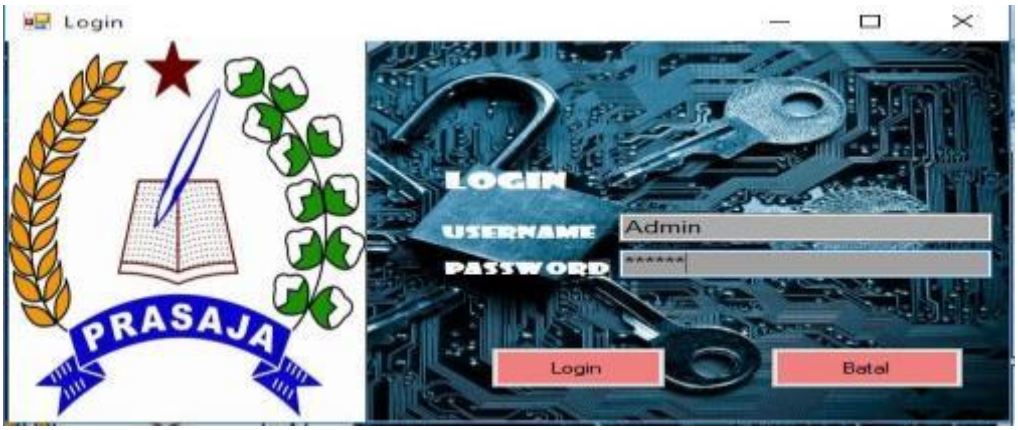
C = {1162, 892, 1320, 65, 1442, 1450, 1162, 1320, 734, 1162, 1245, 65, 1067, 1470, 892, 65, 143, 0}
Z = {AJI SYAIBAN TKJ 90}

4. IMPLEMENTASI DAN PENGUJIAN

Implementasi merupakan tahap dimana aplikasi siap untuk dioperasikan pada keadaan yang sebenarnya sesuai dari hasil analisis dan perancangan yang dilakukan, sehingga akan diketahui apakah sistem atau aplikasi yang dirancang benar-benar dapat menghasilkan tujuan yang dicapai. Aplikasi Sistem Pakar ini dilengkapi dengan *user interface* yang menarik dan bertujuan untuk memudahkan pengguna dalam menggunakannya. Pada aplikasi ini memiliki *interface* atau desain form yang terdiri dari form *Login*, form menu utama, form kerusakan, form gejala, form Basis Aturan, Form Deteksi, dan form laporan.

1. *Form Login*

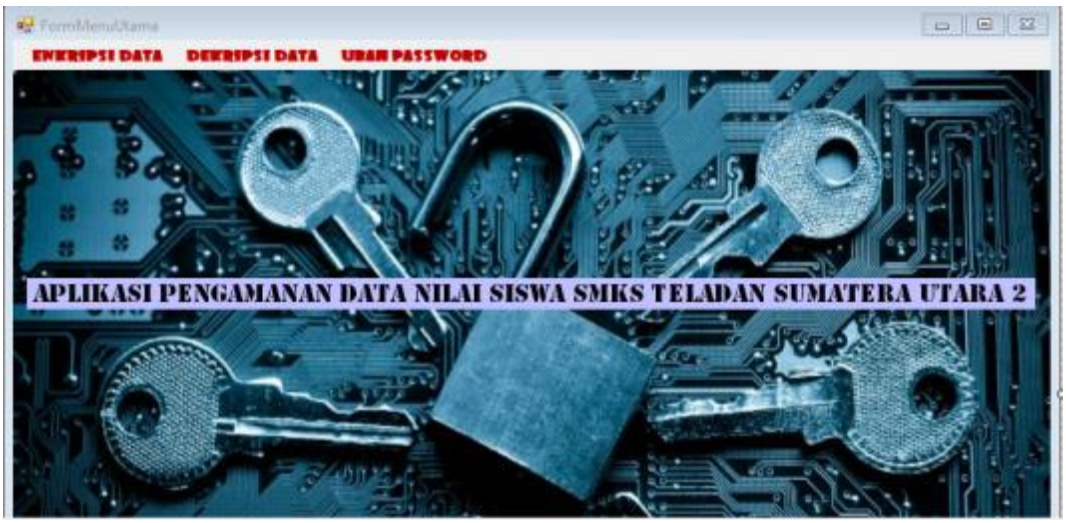
Form Login yang berfungsi untuk pengamana user, apabila di *form* login user memasukkan username dan password dengan benar maka program akan lanjut ke form menu utama, tapi apabila di *form* login user memasukkan username dan password salah akan ada pemberitahuan dari program tersebut*:*



Gambar 5.1 *Form Login*

2. *Form* Menu Utama

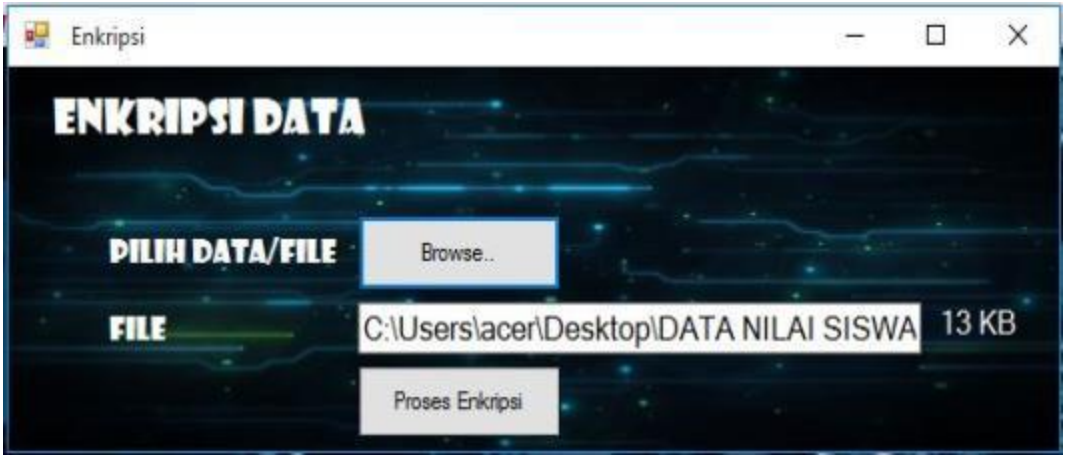
Form Menu Utama berfungsi untuk memudahkan user untuk memanggil form-form yang telah dibuat dalam satu project aplikasi. Berikut tampilan untuk form menu utama:



Gambar 5.2 *Form* Menu Utama

3. *Form* Enkripsi

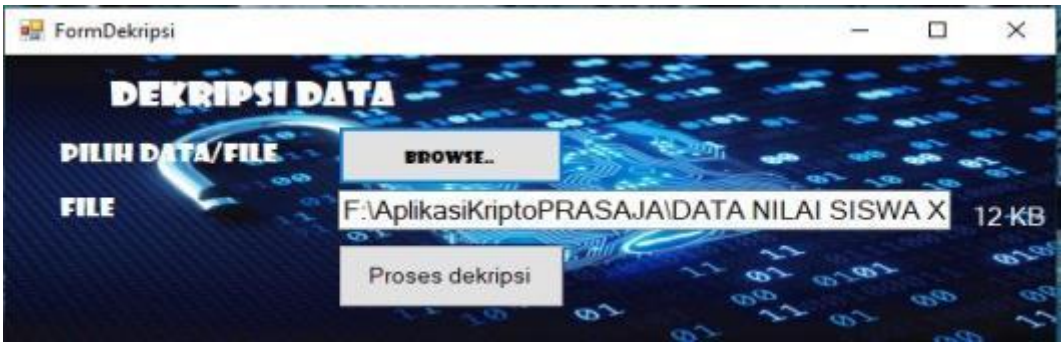
Berfungsi sebagai proses untuk mengamankan Data Nilai Siswa SMKS Teladan dengan cara mengubahnya menjadi kode tertentu. Berikut ini merupakan tampilan dari Form Enkripsi yang berfungsi untuk mengenkripsi data nilai pada SMKS Teladan Sumatera Utara 2:



Gambar 5.3 *Form* Enkripsi

4. *Form* Dekripsi

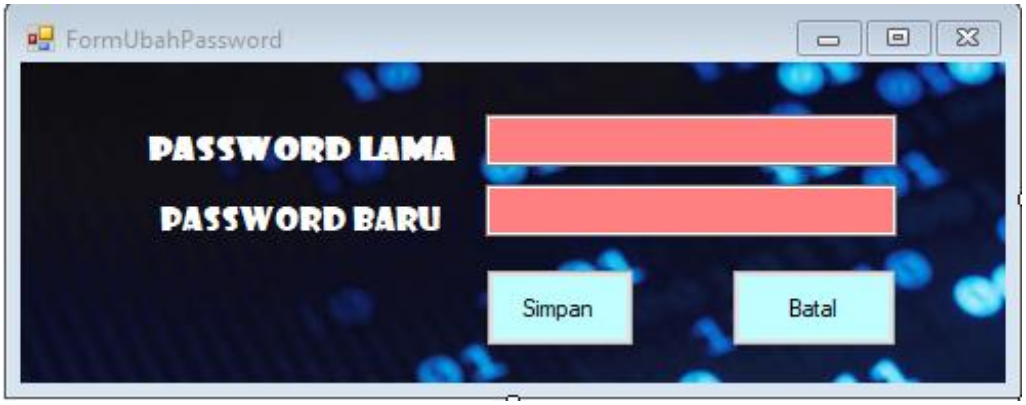
Dekripsi berfungsi untuk pengembalian Data Nilai Siswa yang telah diamankan kembali kedata aslinya. Berikut ini merupakan tampilan dari *Form* Dekripsi yang berfungsi untuk mendekripsi data nilai pada SMKS Teladan Sumatera Utara 2:



Gambar 5.4 *Form* Dekripsi

5. *Form* Ubah Password

Form Ubah Password berfungsi untuk pengubahan password lama menjadi password yang baru. Berikut ini merupakan tampilan dari *Form* Ubah Password:



Gambar 5.5 *Form* Ubah Password

4 Kesimpulan

Berdasarkan perumusan dan pembahasan bab-bab sebelumnya dapat diambil kesimpulan sebagai berikut:

1. Berdasarkan hasil penelitian yang telah dilakukan sebelumnya, metode Merkle Hellman dapat diterapkan kedalam sebuah aplikasi agar dapat mengamankan Data Nilai Siswa dengan baik.
2. Dengan mengimplementasikan algoritma Merkle Hellman pada sistem pengamanan data, dapat mengenkripsi atau mengubah pesan menjadi bentuk kode-kode yang tidak dapat dimengerti oleh siapapun.
3. Dalam merancang aplikasi menggunakan Merkle Hellman yang dapat digunakan dalam pengamanan data nilai siswa, yaitu dengan membuat pemodelan sistem seperti *use case diagram*, *activity diagram* dan *class diagram*, kemudian membuat *flowchart* algoritma sistem, dan terakhir melakukan pengkodean dengan program VB 2008.
4. Sistem yang telah dirancang selanjutnya diuji dan diimplementasikan dengan memasukkan data-data sesuai dengan yang ada pada bab-bab sebelumnya, kemudian jika hasil outputnyya sesuai dengan data manual maka dalam pengujian ini sistem berjalan dengan baik.

UCAPAN TERIMA KASIH

Puji Syukur kepada Tuhan Yang Maha Esa atas karunia-Nya, dengan kasih sayang dan kekuatan-Nya dalam menyelesaikan karya tulis ini sebagai skripsi dengan judul: "Penerapan Metode Merkle Hellman Untuk Pengamanan Data Nilai Siswa Pada SMKS Teladan Sumatera Utara 2". dapat diselesaikan dengan tepat pada waktu yang telah ditentukan. Terima kasih tak terhingga kepada kedua orang tua tercinta Bapak Poltak Berutu dan Ibu Lince Sipahutar yang telah memberikan doa dan dukungan baik secara moral maupun materil sehingga mampu menyelesaikan pendidikan dari tingkat sekolah dasar sampai bangku perkulihaan dengan baik.

REFERENSI

[1] B. A. B. Ii, T. Sistem, and P. Di, "Bab ii tinjauan tentang sekolah menengah atas 2.1.," pp. 11–37, 2008.

[2] A. N. Agustina, "PENGAMANAN DOKUMEN MENGGUNAKAN METODE RSA (RIVEST SHAMIR ADLEMAN) BERBASIS WEB," pp. 14–19, 2015.

[3] J. I. Mulawarman *et al.*, "IMPLEMENTASI KRIPTOGRAFI PENGAMANAN DATA PADA PESAN TEKS , ISI FILE DOKUMEN , DAN FILE DOKUMEN MENGGUNAKAN ALGORITMA ADVANCED ENCRYPTION," vol. 10, no. 1, 2015.

[4] M. Simanjuntak, "Implementasi Algoritma Merkle Hellman untuk Keamanan Database," vol. 4, no. 1, pp. 46–50, 2019.

[5] F. Zuli and A. Irawan, "Implementasi Kriptografi Dengan Algoritma Blowfish dan Riverst Shamir Adleman (RSA) Untuk Proteksi File," vol. 6, pp. 27–38, 2016.

[6] P. Studi, T. Informatika, F. Ilmu, K. Universitas, and D. Bengkulu, "APLIKASI KRIPTOGRAFI PESAN MENGGUNAKAN ALGORITMA," vol. 10, no. 2, pp. 120–128, 2014.

BIOGRAFI PENULIS

	**Imorananta Berutu** Wanita kelahiran Bongkaras, 15 Juli 1998 anak ke 3 dari 4 bersaudara pasangan Bapak Poltak Berutu dan ibu Lince Sipahutar, Mempunyai pendidikan Sekolah Dasar SD Negeri 037155 Bongkaras tamat tahun 2010, kemudian melanjutkan pendidikan Sekolah Menengah Pertama SMP Negeri 1 Silima Punggapungga 2013, kemudian melanjutkan pendidikan Sekolah Menengah Atas SMA Negeri 1 Silima Punggapungga tamat tahun 2016. Saat ini menempuh pendidikan Strata Satu (S-1) di STMIK Triguna Dharma Medan mengambil jurusan Program Studi Sistem Informasi. E-mail ranantabrutu@gmail.com
	**Badrul Anwar, SE, S.Kom., M.Kom** Beliau merupakan dosen tetap STMIK Triguna Dharma, serta aktif sebagai dosen pengajar khusus pada bidang ilmu Sistem Informasi.
	**Tugiono, S.Kom.,M.Kom** Beliau merupakan dosen tetap di STMIK Triguna Dharma serta aktif sebagai dosen pengajar khusus di bidang ilmu Sistem Informasi.