Jurnal CyberTech
Vol.4. No.2, Februari 2021, pp. 01-14
P-ISSN : 9800-3456
E-ISSN : 2675-9802

IMPLEMENTASI KRIPTOGRAFI UNTUK MENGAMANKAN DATA ABSENSI KARYAWAN PADA PT. PENERBIT ERLANGGA MAHEMERU MEDAN MENGGUNAKAN (DATA ENCRYPTION STANDARD)

Arief Anrico Pandiangan *,Nurcahyo Budi Nugroho, S.Kom., M.Kom, Afdal Alhafiz S.Kom., M.Kom****

* Sistem Informasi, STMIK Triguna Dharma
** Program Studi Dosen Pembimbing, STMIK Triguna Dharma

Article Info
Article history:
Received Feb 12th, 2021
Revised Feb 20th, 2021
Accepted Feb 26th, 2021

Keyword:
First keyword
Second keyword
Third keyword
Fourth keyword
Fifth keyword

ABSTRACT

Dalam sebuah perusahaan Data absensi karyawan dibutuhkan untuk mengetahui apakah seorang karyawan datang dan meninggalkan kantor sesuai dengan jam kerja yang telah di tentukan atau tidak. selain itu absensi juga dibutuhkan untuk mengetahui apakah seorang karyawan bekerja lembur atau tidak, dan oleh sebab itu PT. erlangga juga harus lebih meningkatkan keamanan data absensi harian karyawan agar tidak. Data Encryption Standard (DES) adalah salah satu metode kriptografi cipher blok yang popular digunakan karena tingginya tingkat keamanan informasi dan dijadikan standard algoritma enkripsi kunci-simetri. DES adalah nama standard enkripsi simetri yang dahulu memiliki nama algoritma enkripsinya DEA (Data Encryption Algorithm), namun nama DES lebih populer dari pada DEA.

Corresponding Author: *First Author
Nama :Arief Anrico Pandiangan
Program Studi
STMIK Triguna Dharma
Email: Ariefpandiangan11@gmail.com

1. PENDAHULUAN

Berdiri pada 30 April 1952, Erlangga Group pada mulanya hanya menerbitkan buku-buku pelajaran saja. Namun kini, di usia yang ke 68, PT. penerbit erlangga telah menduduki posisi

mapan di ranah penerbitan Indonesia PT.Penerbit erlangga semakin dikenal karena tingginya kualitas dan kayanya ragam buku yang diterbitkan[1].

Tertempa oleh berpuluh tahun jatuh dan bangunnya sektor pendidikan di Indonesia, dalam skala nasional kami adalah penerbit buku pelajaran yang terbaik. Di luar itu, sejak sepuluh tahun yang lalu Erlangga Group mulai melakukan pengembangan usaha dengan menerbitkan judul-judul buah karya penulis yang dikenal di ranah nasional maupun internasional, baik untuk buku anak maupun buku populer. Erlangga Group adalah rumah bagi buku pelajaran terbaik, penulis-penulis yang ternama, dan juga karakter kartun kesayangan anak.

Dalam sebuah perusahaan Data absensi karyawan dibutuhkan untuk mengetahui apakah seorang karyawan datang dan meninggalkan kantor sesuai dengan jam kerja yang telah di tentukan atau tidak. selain itu absensi juga dibutuhkan untuk mengetahui apakah seorang karyawan bekerja lembur atau tidak, dan oleh sebab itu PT. erlangga juga harus lebih meningkatkan keamanan data absensi harian karyawan agar tidak[2].

terjadi hal - hal yang tidak diinginkan Maka di butuhkan sebuah metode penyandian, ilmu sekaligus seni guna menjaga informasi yang disebut juga dengan Kriptografi. Untuk mengatasi permasalahan tersebut, penulis membuat program aplikasi pengamanan data absensi karyawan dengan menggunakan teknologi informasi yang berbasis desktop yaitu kriptografi.

Kriptografi merupakan seni dan ilmu untuk memproteksi pengiriman data dengan cara mengubah nya menjadikan kode tertentu dan hanya di tujukan untuk orang yang hanya memiliki sebuah kunci untuk mengubah kode itu kembali yang berfungsi untuk menjaga kerahasiaan data atau pesan. dalam kriptografi, data atau pesan yang di kirimkan melalui jaringan akan di samarkan sedemikian rupa. Sehingga seandai nya data tersebut dapat di peroleh dan di baca oleh orang lain, maka pihak yang tidak berhak dan berwenang tidak akan dapat mengerti arti dari data tersebut[3].

dalam bidang kriptografi terdapat dua konsep yang sangat penting atau utama yaitu Enskripsi dan Dekripsi .Enskripsi adalah proses dimana informasi atau data yang hendak di kirim di ubah menjadi bentuk yang hampir di kenal sebagai informasi awal nya dengan menggunakan algoritma tertentu.Dekripsi adalah kebalikan dari enkripsi yaitu mengubah kembali bentuk yang tersamar tersebut menjadi informasi awal.sebuah pesan atau data yang masih asli dan belum mengalami penyandian di kenal dengan isitilah *plaintext* [4] . Salah satu metode kriptografi yang digunakan dalam pengamanan data absensi adalah metode des (Data Encryption Standard)

DES *(Data Enycription Standard)* pengguna data sandi yang paling banyak

di dasarkan pada dasar standard standard data sandi DES yang di ambil pada tahun 1977 oleh standard standard nasional Bureau, yang sekarang Institut Nasional Standard dan teknologi (NIST),sebagai standard informasi umum .untuk DES, data di sandikan kedalam 64 balok bit menggunkan 56 bit kunci.Transformasi algoritma 64 bit *input* ke dalam satu langkah langkah ke dalam 64 bit output.langkah pertama dengan kunci yang sama digunakan untuk cadangan persandiaan[5].

Dari pembahasan penelitiann tersebut di harapkan *software* yang di rancang dapat membantu pihak PT. penerbit erlangga mahemeru medan dalam mengamankan data absensi karyawan nya menggunakan kunci Asimetris berdasarkan deksripsi masalah di atas maka dilakukan penilitian skripsi dengan judul **"IMPLEMENTASI KRIPTOGRAFI UNTUK MENGAMANKAN DATA ABSENSI KARYAWAN PADA PT. PENERBIT ERLANGGA MAHEMERU MEDAN MEN GUNAKAN METODE *DES (DATA ENCRYPTION STANDARD)"***

2. METODE PENELITIAN

Metode penelitian merupakan pengumpulan informasi terhadap objek yang akan diteliti serta melakukan investigasi pada data yang didapatkan tersebut. Dalam melakukan penelitian diharuskan untuk terjun langsung kelapangan untuk mendapatkan data sesuai dengan yang akan diteliti.

2.1 PENGUMPULAN DATA

1. Observasi
   Observasi merupakan salah satu teknik dalam pengumpulan data yang kompleks. Dalam penelitian ini observasi dilakukan untuk mendapatkan data di PT. Penerbit Erlangga. Hal ini bertujuan untuk memperoleh informasi tentang data yang akan digunakan dalam penelitian ini. Data yang didapatkan di PT. Penerbit Erlangga Mahemeru Medan adalah data absensi karyawan.
2. Wawancara
   Teknik wawancara ini dilakukan untuk mendapatkan informasi tambahan dari pihak-pihak yang memiliki wewenang dan berinteraksi langsung dengan dengan karyawan PT. Erlangga (Juanda Pandiangan).

Tabel 3.1 Data Absensi Karyawan

NIK	Nama Karyawan	Tanggal	Jam Masuk	Jam Keluar	Keterangan
1166754267788	Prayitno	13/12/2020	9:30:00 AM	4:00:00 PM	-

2.2. Studi Kepustakaan (Study of Literature)

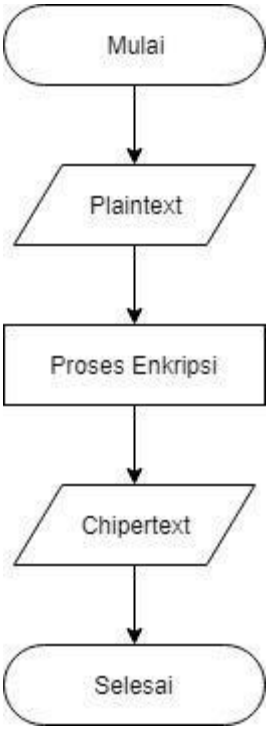
Dalam penelitian ini banyak menggunakan jurnal-jurnal baik jurnal nasional maupun buku sebagai sumber referensi. Dari komposisi yang ada jumlah literatur yang digunakan sebanyak 22 dengan rincian: 21 jurnal nasional, dan 1 buku nasional.

2.2. Algoritma Sistem

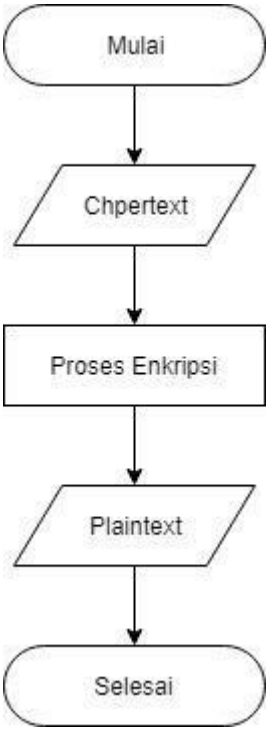
Algoritma sistem merupakan penjelasan langkah-langkah dalam penyelesaian masalah dalam perancangan sistem keamanan data nasabah dengan menggunakan algoritma DES. Hal ini dilakukan untuk meningkatkan keamanan data nasabah tersebut.

2.3 Flowchart Dari Metode Penyelesaian

Berikut ini adalah flowchart dari proses enkripsi dan dekripsi dari algoritma DES yaitu sebagai berikut:



Gambar 3. 1 Flowchart *Proses Enkripsi*



Gambar 3. 2 *Flowchart* proses dekripsi

2.4 Dekripsi Data Penelitian

Berikut ini adalah data nasabah yang di dapat dari koperasi sinode parbubu, yang akan diamankan. Dalam pengujiannya, sebagai contoh data yang digunakan sebagai sampel dalam penelitian ini yaitu sebagai berikut:

Tabel 3.2 Data Absensi Karyawan

NIK	Nama Karyawan	Tanggal	Jam Masuk	Jam Keluar	Keterangan
1166754267788	Prayitno	13/12/2020	9:30:00 AM	4:00:00 PM	-

2.4.1 Proses Enkripsi

Proses enkripsi adalah mengubah suatu data plaintext ke chippertext. Dalam proses enkripsi terdapat beberapa langkah-langkah berikut:

1. Mengubah *Plaintext* dan *Key* Menjadi Bilangan Biner

Mengubah *plaintext* kedalam biner berdasarkan tabel ASCII.

Tabel 3. 2 Konversi plaintext *ke biner*

PLAINTEXT			
	DEC	hexa	biner
P	80	50	01010000
r	114	72	01110010
a	97	61	01100001
y	121	79	01111001
i	105	69	01101001
t	116	74	01110100
n	110	6E	01101110
o	111	6F	01101111

Mengubah *key* ke dalam biner berdasarkan tabel ASCII

Tabel 3.3 Konversi key *ke biner*

KEY			
	DEC	HEXA	BINER
A	65	41	01000001
R	82	52	01010010
I	73	49	01001001
E	69	45	01000101
F	70	46	01000110
1	49	31	00110001
2	50	32	00110010
3	51	33	00110011

2. Initial Permutation Plaintext

Lakukan *initial permutation* (IP) pada bit plaintext menggunakan tabel IP seperti berikut:

Tabel 3. 4 Initial permutation

PLAINTEXT (X)							
0	1	0	1	0	0	0	0
0	1	1	1	0	0	1	0
0	1	1	0	0	0	0	1
0	1	1	1	1	0	0	1
0	1	1	0	1	0	0	1
0	1	1	1	0	1	0	0
0	1	1	0	1	1	1	0
0	1	1	0	1	1	1	1

IP1							
58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6
64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1
59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7

Keterangan pada tabel *initial permutation* dan tabel IP(X):

Angka 0 dan 1 merupakan bilangan biner

Angka **1,2,3** dan seterusnya yang menggunakan penebalan adalah urutan posisi bit

Urutan bit ke-58 pada tabel *plaintext* (X), diletakan pada posisi 1 pada tabel IP,

Urutan bit ke-50 pada tabel *plaintext* (X), diletakan pada posisi 2 pada tabel IP,

Urutan bit ke-42 pada tabel *plaintext* (X), di letakan pada posisi 3 pada tabel IP,

Demikian seterusnya dan menghasilkan Tabel IP(X).

Tabel 3. 5 IP(X)

Tabel IP(X)								
1	1	1	1	1	1	1	1	**L0**
0	0	1	0	1	0	1	1	
1	1	1	0	0	0	0	0	
1	0	0	1	1	1	0	0	
0	0	0	0	0	0	0	0	**R0**
1	1	1	1	1	1	1	0	
1	1	0	1	1	0	0	0	
1	1	0	0	0	0	1	0	

Selanjutnya bit pada IP(X) di pecah menjadi dua bagian yaitu L0 dan R0 sehingga hasilnya dapat di lihat pada tabel 3.5.

3. Melakukan Permutasi *Key* Kompresi PC-1

Kunci yang sudah diubah menjadi bilangan biner, lalu di permutasikan dengan menggunakan tabel permutasi kompresi PC-1, pada langkah ini terjadi kompresi 64 bit menjadi 56 bit.

Tabel 3. 6 Permutasi Kompresi PC-1

KEY							
0	1	0	0	1	1	0	1
0	1	0	0	0	0	0	1
0	1	0	1	0	0	1	0
0	1	0	0	1	0	0	1
0	1	0	0	1	1	1	1
0	1	0	0	0	1	0	1
0	1	0	0	1	0	1	1
0	1	0	0	1	1	1	1

PC1						
57	49	41	33	25	17	9
1	58	50	42	34	26	18
10	2	59	51	43	35	27
19	11	3	60	52	44	36
63	55	47	39	31	23	15
7	62	54	46	38	30	22
14	6	61	53	45	37	29
21	13	5	28	20	12	4

Keterangan pada tabel Permutasi Kompresi PC-1

Angka 0 dan 1 merupakan bilangan biner

Angka **1,2,3** dan seterusnya yang menggunakan penebalan adalah urutan posisi bit

Urutan bit ke-57 pada tabel *key*, diletakan pada posisi 1 pada Tabel PC-1,

Urutan bit ke-49 pada tabel *key*, diletakan pada posisi 2 pada Tabel PC-1 dst, dan hasil permutasi *key* dapat di lihat pada tabel 3.8.

Tabel 3. 7 PC-1

TABEL PC-1							
0	0	0	0	0	0	0	C0
0	0	0	0	1	1	1	
1	1	1	1	1	0	0	
0	0	0	1	1	1	0	
1	1	0	1	0	0	1	D0
0	0	0	0	1	1	0	
0	0	0	0	0	0	0	
1	0	0	0	0	1	0	

Selanjutnya bit pada Tabel hasil permutasi PC-1 di pecah menjadi dua bagian yaitu C0 dan D0 sehingga hasilnya sebagai berikut.
C0: 0000000 0000111 1111100 0001110
D0: 1101001 0000110 0000000 1000010
4. Melakukan Pergeseran Kiri

Lakukan pergeseran kiri (*left Shift Operation*) pada C0 dan D0 sebanyak satu atau dua kali berdasarkan putaran yang ada pada tabel putaran sebagai berikut:

Tabel 3. 8 Left shif

Putaran ke – i	**Jumlah Pergeseran Bit (Left Shift)**
1	1
2	1
3	2
4	2
5	2
6	2
7	2
8	2
9	1
10	2
11	2
12	2
13	2
14	2
15	2
16	1

Keterangan:
Untuk putaran ke-1, dilakukan pergeseran 1bit ke kiri,
Untuk putaran ke-2, dilakukan pergeseran 1 bit ke kiri,
Untuk putaran ke-3, dilakukan pergeseran 2 bit ke kiri, dan seterusnya hingga putaran yang ke-16.
Berikut hasil dari *left shift*:
Putaran ke-1, di geser 1 bit ke kiri.
C1: 0000000 0001111 1111000 0011100
D1: 1010010 0001100 0000001 0000101
Putaran ke-16, di geser 1 bit ke kiri.
C16: 0000000 0000111 1111100 0001110
D16: 1101001 0000110 0000000 1000010
Setiap hasil putaran digabungkan kembali menjadi C_iD_i dan diinput kedalam tabel *Permutation Compression* 2 (PC-2) dan terjadi kompresi data C_iD_i 56 bit menjadi C_iD_i 48 bit dan mengkasilkan K_i.

Tabel 3. 9 Permutation Compression *2 (PC-2)*

PC-2					
14	17	11	24	1	5
3	28	15	6	21	10
23	19	12	4	26	8
16	7	27	20	13	2
41	52	31	37	47	55
30	40	51	45	33	48
44	49	39	56	34	53
46	42	50	36	29	32

Keterangan:
Urutan bit pada C_iD_i yang ke-14, diletakan di posisi 1 pada tabel PC-2,
Urutan bit pada C_iD_i yang ke-17, diletakan di posisi 2 pada tabel PC-2,
Urutan bit pada C_iD_i yang ke-11, diletakan di posisi 3 pada tabel PC-2, dan seterusnya.

Berikut hasil *output*nya:
K1 111100001000001010100010001000010000011110000010
K2 111100001001101000100010000001100011001101000000
K15 110000011000010010101110101000101010000100010100
K16 110000001000011010101010100001110000000000001011

5. Melakukan Ekspansi Data

Pada langkah ini, kita akan meng-ekspansi data R_{i-1} 32 bit menjadi R_i 48 bit sebanyak 16 kali putaran dengan nilai perputaran 1>= i <=16 menggunakan Tabel Ekspansi (E).

Tabel 3. 10 Ekspansi

Tabel Ekspansi					
32	1	2	3	4	5
4	5	6	7	8	9
8	9	10	11	12	13
12	13	14	15	16	17
16	17	18	19	20	21
20	21	22	23	24	25
24	25	26	27	28	29
28	29	30	31	32	1

Hasil $E(R_{i-1})$ kemudian di X OR dengan K kan Vektor Matriks A_i. Berikut hasil *output*nya:

Iterasi 1

E((R1)-1) 000000 000001 011111 111101 011011 110001 011000 000100
K1 111100 001000 001010 100010 001000 010000 011110 000010
---------- XOR
A1 111100 001001 010101 011111 010011 100001 000110 000110

Pada iterasi satu (1) diatas didapat A1 dari hasil XOR E(R1-1) dan K1, setelah itu maka proses selanjutnya langsung ke langkah ke-6 terlebih dahulu, dimana A_i, akan dimasukan ke dalam S-BOX dan menghasilkan PB1 yang kemudian di XOR kan dengan L0 dan menghasilkan nilai R_i. Nilai R_i ini digunakan untuk melanjutkan

iterasi ke-2.

6. Memasukan Data ke S-BOX

A1 111100 001001 010101 011111 010011 100001 000110 000110

Tabel 3. 11 Subtitusi S1

	0000	**0001**	**0010**	**0011**	**0100**	**0101**	**0110**	**0111**	**1000**	**1001**	**1010**	**1011**	**1100**	**1101**	**1110**	**1111**
00	14	4	13	1	2	15	11	8	3	10	6	12	5	9	0	7
01	0	15	7	4	14	2	13	1	10	6	12	11	9	5	3	8
10	4	1	14	8	13	6	2	11	15	12	9	7	3	10	5	0
11	15	[redacted]													6	13

Kemudian kita ambil sampel blok bit pertama yaitu **001000**, pisahkan menjadi 2 blok yaitu:

1. Bit pertama dan terakhir yaitu 1 dan 0, digabungkan menjadi 10
2. Bit kedua hingga kelima yaitu 1110

Selanjutnya dibandingkan dengan memeeriksa perpotongan antara kedua di dapatkan nilai 5 (warna kuning) lalu dibinerkan menjadi **0101**

Tabel 3. 12 Subtitusi S2

	0000	**0001**	**0010**	**0011**	**0100**	**0101**	**0110**	**0111**	**1000**	**1001**	**1010**	**1011**	**1100**	**1101**	**1110**	**1111**
00	15	1	8	14	6	11	3	4	9	7	2	13	12	0	5	10
01	3	13	4	7	15	2	8	14	12	0	1	10	6	9	11	5
10	0	14	7	11	10	4	13	1	5	8	12	6	9	3	2	15
11	13	8	10	1	3	15	4	2	11	6	7	12	0	5	14	9

Kemudian kita ambil sampel blok bit kedua yaitu **001000**, pisahkan menjadi 2 blok yaitu:

1. Bit pertama dan terakhir yaitu 0 dan 1, digabungkan menjadi 01
2. Bit kedua hingga kelima yaitu 0100

Selanjutnya dibandingkan dengan memeeriksa perpotongan antara kedua di dapatkan nilai 15 (warna kuning) lalu dibinerkan menjadi **1111.** Dan seterusnya untuk blok ketiga hingga blok kedelapan dibandingkan dengan S3 dan S8. Berdasrkan cara diatas diperoleh hasil sebagai berikut:

B1 = 01011111 01011001 00000100 11100100

7. Memutasikan Bit Vektor Bi

Setelah didapatkan nilai vector B_i, langkah selanjutnya adalah memutasikan bit vektor B_i menggunakan tabel P-*BOX*, lalu dikelompokkan menjadi 4 blok dimana setiap blok memiliki 32 bit data

Tabel 3. 13 Matrik Permutasi P (P-box)

P-BOX							
16	7	20	21	29	12	28	17
1	15	23	26	5	8	31	10
2	8	24	14	32	27	3	9
19	13	30	6	22	11	4	25

Sehingga hasil yang didapatkan sebagai berikut:

P(B1) : 01101110 01110101 00010010 01010101

Hasil P(B_i) kemudian di XOR kan dengan L_i-1 untuk mendapatkan nilai R_i. Sedangkan nilai L_i sendiri diperoleh dari nilai R_i-1 untuk nilai 1>=i<=16.

L0 : 11111111 00101011 11100000 10011100
R0 : 00000000 11111110 11011000 11000010
P(B1) : 11000100 00011001 11000100 01111011
L0 : 11111111 00101011 11100000 10011100
---------- XOR
R1 : 00111011 00110010 00100100 11100111

Untuk mencari R2 sampai R16, lakukan langkah yang sama dari langkah 5 sampai 7 dan dituliskan dalam bentuk itrasi, sehingga pada iterasi ke-16 didapatkan hasil sebagai berikut.

Iterasi 16

E (R15) : 10101010101111011010110000100000000000001000000110
K16 : 11000000100001101010101010000111000000000000001011
---------- XOR
A16 : 01101010001110110000011010100111000000010000001101
B16 : 10011000001100110001011111110111
P(B16) : 10100110111110100010110000101111
L(16)-1 : 11011000111000101001010011001000
---------- XOR
R16 : 01111110000110001011100011100111
L(16) : 11011000111000101001010011001000

8. Menggabungkan R16 dan L16

Langkah terakhir adalah menggabungkan R_{16} dengan L_{16} kemudian dipermutasikan dengan tabel *initial permutation* (IP-^{1}).

Tabel 3. 14 Permutasi R16 dan L16 dengan Tabel IP^{-1}

R16 dan L16								
R16	0	1	1	1	1	1	1	0
	0	0	0	1	1	0	0	0
	1	0	1	1	1	0	0	0
	1	1	1	0	0	1	1	1
L16	0	1	0	1	0	1	0	1
	1	0	1	1	0	1	1	0
	0	1	0	0	0	0	0	0
	0	1	0	0	0	0	1	1

Tabel IP-1							
40	8	48	16	56	24	64	32
39	7	47	15	55	23	63	31
38	6	46	14	54	22	62	30
37	5	45	13	53	21	61	29
36	4	44	12	52	20	60	28
35	3	43	11	51	19	59	27
34	2	42	10	50	18	58	26
33	1	41	9	49	17	57	25

Tabel 3. 15 Chipertext

CHIPERTEKS							
1	0	0	0	0	0	1	1
0	1	1	0	0	0	1	1
1	1	1	0	0	0	0	1
0	1	0	1	0	1	0	0
1	1	1	1	0	1	0	0
0	1	1	0	0	1	0	1
1	1	0	0	1	0	1	1
0	0	1	0	0	1	0	1

Menghasilkan *output*:
Chiper dalam biner : **10000011 01100011 11100001 01010100 11110100 01100101 11001011 00100101**
Atau dalam *chiper* hexa : **83 63 E1 54 F4 65 CB 25**

3.3.4 Proses Dekripsi

Untuk dapat mengetahui isi pesan sebenarnya, perlu dilakukan konversi *ciphertext* menjadi bentk biner untuk mendapatkan bit *chipertext*. Dekripsi dapat dilakukan sebagai berikut:

1. Melakukan Permutasi Terhadap *Chiper*

Chiper dalam biner : **10000011 01100011 11100001 01010100 11110100 01100101 11001011 00100101**
Atau dalam *chiper* hexa : **83 63 E1 54 F4 65 CB 25**

Tabel 3. 16 Initial permutation chipper *(IP)*

Ciphertext							
1	0	0	0	0	0	1	1
0	1	1	0	0	0	1	1
1	1	1	0	0	0	0	1
0	1	0	1	0	1	0	0
1	1	1	1	0	1	0	0
0	1	1	0	0	1	0	1
1	1	0	0	1	0	1	1
0	0	1	0	0	1	0	1

Tabel IP							
58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6
64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1
59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7

Tabel 3.17 Hasil *initial permutation chipper* (IP)

IP(Cipher)								
0	1	1	1	1	1	1	0	L0
0	0	0	1	1	0	0	0	
1	0	1	1	1	0	0	0	
1	1	1	0	0	1	1	1	
0	1	0	1	0	1	0	1	R0
1	0	1	1	0	1	1	0	
0	1	0	0	0	0	0	0	
0	1	0	0	0	0	1	1	

Selanjutnya bit pada IP (*Chiper*) dipecah menjadi 2 bagian yaitu L0 dan R0,
Sehingga menghasilkan sebagai berikut:
L0 : 01111110000110001011100011100111
R0 : 01010101101101100100000001000011

Iterasi 16

P(B16) : 10100110111110100010110000101111
L15 : 01111110000110001011100011100111
---------- XOR
R16 : 11011000111000101001010011001000

Lakukan iterasi 15-1 sehingga didapatkan, pada iterasi pertama sebagai berikut.

Iterasi 1

P(B1) : 01001010 00000101 00011111 10011100
L0 : 01001010 11111010 00011111 10011001
---------- XOR
R1 : 11111111000000010101010010101110
L1 : 00000000111111101100010001000001

2. Melakukan Permutasi R1 dan L1 dengan Tabel IP-1

Kemudian R_1 dan L_1 di permutasikan kembali dengan tabel *inverse initial permutation* sehingga menghasilkan *output*:

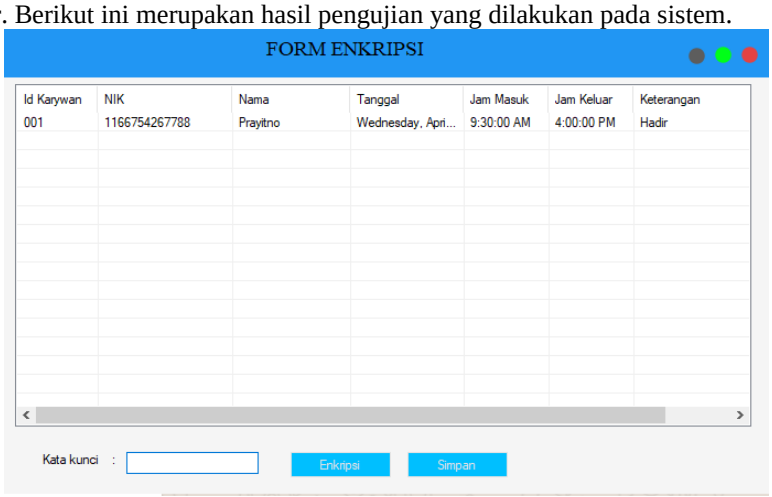
Plaintext dalam biner : **01010000 01110010 01100001 01111001 01101001 01110100 01101110 01101111**
Atau dalam bentuk hexa: **50 72 61 79 69 74 6E 6F**

3. ANALISA DAN HASIL

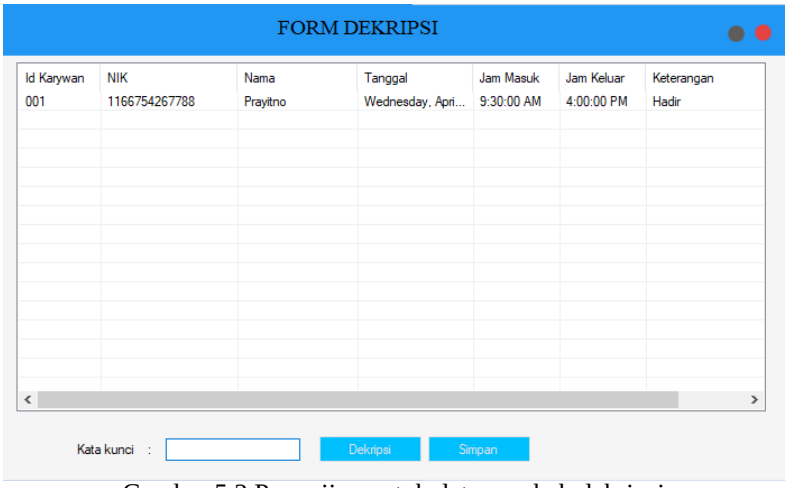
3.1 Pengujian Sistem

Uji coba sistem bertujuan untuk membuktikan bahwa *input*, *proses*, *output* yang dihasilkan oleh sistem aplikasi *Visual Studio* 2012 telah benar dan sesuai dengan yang diinginkan. Pengujian sistem dengan cara memasukkan data ke dalam sistem dan memperhatikan *output* yang dihasilkan. Jika *input*, proses dan *output* telah sesuai, maka sistem telah benar. Berikut merupakan tahapan untuk pengujian sistem yaitu:

1. Melakukan *input* data nasabah yang kemudian sistem akan menampilkan data nasabah yang tersimpan di *database*.
2. Menggunakan bahasa pemrograman *Microsoft Visual Studio* 2012 dalam pengolahan data yang disimpan dalam *database Microsoft Office Access* 2010 . Penggunaan sistem pengamanan data nasabah pada PT.Erlangga Mahemeru Medan, agar dapat berjalan dengan baik *file* aplikasi *Visual Studio* 2012 harus ditempatkan pada satu *folder* dan dilengkapi dengan *input* data dari analisa sistem. Lokasi *folder* yang telah ditentukan adalah tempat untuk menyimpan *file-file* yang telah dikumpulkan, untuk menghindari kesalahan sebaiknya data tidak diletakkan kedalam *folder* yang berbeda. Selanjutnya untuk menerapkan metode dalam mengamankan data nasabah, maka data tersebut akan *diinput* ke aplikasi lalu simpan data tersebut ke dalam *database Access*. Jalankan aplikasi *Visual Studio* 2012 yang telah terinstal dikomputer. Berikut ini merupakan hasil pengujian yang dilakukan pada sistem.



Gambar 5.1 Pengujian untuk data nasabah enkripsi



Gambar 5.2 Pengujian untuk data nasabah dekripsi

4. KESIMPULAN

Berdasarkan pembahasan dan evaluasi dari bab sebelumnya, maka dapat ditarik kesimpulan sebagai berikut :

1. Dalam menganalisa masalah yang terjadi terkait dengan pengamanan data absensi di PT. Penerbit Erlangga Mahemeru Medan menggunakan algotritma *Data Encryption Standard* (DES) maka dilakukan proses enkripsi untuk data absensi.
2. Perancang sistem kriptografi yang mengadopsi algoritma DES (*Data Encryption Standard)* dengan metode sistem *Block Cipher* di dalam menyelesaikan masalah terkait pengamanan data absensi di PT. Penerbit Erlangga Mahemeru Medan menggunakan pemrograman yang berbasis desktop.
3. Pengujian sistem ini dilakukan sebelum nantinya dapat dicoba untuk membantu instansi-instansi terkait di dalam pengamanan data di PT. Penerbit Erlangga Mahemeru Medan.

UCAPAN TERIMA KASIH

Terimakasih buat sahabat yang selalu menemani dan memberikan suport kepada saya selama dalam menyelesaikan artikel ilmiah ini terkhusus buat teman-teman GMKI yang tiada henti-hentinya memberikan masukan-masukan dan saran dalam penyusunan karya ilmiah ini.

REFERENSI

[1] www.penerbit erlangga.com.

[2] D. Adhar, “Implementasi Algoritma Des (Data Encryption Standard) Pada Enkripsi Dan Deskripsi Sms Berbasis Android,” *J. Tek. Inform. Kaputama*, vol. 3, no. 2, pp. 53–60, 2019, [Online]. Available: https://jurnal.kaputama.ac.id/index.php/JTIK/article/view/185.

[3] E. S. Han and A. goleman, daniel; boyatzis, Richard; Mckee, “Peranan Kriptografi Sebagai Keamanan Sistem Informasi Pada Usaha Kecil Dan Menengah,” *J. Chem. Inf. Model.*, vol. 53, no. 9, p. 2, 2019.

[4] Y. Permanasari, “Kriptografi Klasik Monoalphabetic,” vol. 16, no. 1, pp. 7–10, 2017.

[5] A. P. Wahyadyatmika, R. R. Isnanto, and M. Somantri, *Implementasi Algoritma Kriptografi RSA pada Surat Elektronik (E-Mail)*, vol. 3, no. 4. 2014.

BIBLIOGRAFI PENULIS

<table>
<tr><td>  </td><td>Nama : Arief Anrico Pandiangan
Tempat Lahir : Saribudolok
Tanggal Lahir : 02 Februari 1999
Jenis Kelamin : Laki-Laki
Agama : Kristen
Warga Negara : Indonesia
Status : Lajang
Alamat : Jl. Tanjung Anom</td></tr>
<tr><td>  </td><td>Nurcahyo Budi Nugroho, S.Kom., M.Kom</td></tr>
<tr><td>  </td><td>Afdal Alhafiz, S.Kom., M.Kom
NIDN: 0114059301
Program Studi : Sistem Informasi
Tanggal Lahir :Medan 14 Mei 1993
Jabatan Fungsional : Asisten Ahli
Deksripsi :Dosen Tetap STMIK TRIGUNA DHARMA yang aktif dan fokus bidang di bidang keilmuan sistem kendali
Alamat Email:afdal.alhafizh@gmail.com

Riwayat Pendidikan :

1. S1 STMIK Triguna dharma 2014

2.S1 Universitas putra indonesia padang 2018</td></tr>
</table>