Implementasi Metode Kriptografi International Data Encryption Algorithm (IDEA) Untuk Pengamanan Data Berita Publik Khatulistiwa Televisi Bontang

Rosmasari[*1], Rizky Ariesta Dwi RA[2], Nataniel Dengen[3], Medi Taruk[4]
[*1,2,3,4]Jurusan Teknologi Informasi dan Komunikasi, Universitas Mulawarman, Samarinda
e-mail: [*1]rosmasari.unmul@gmail.com, [2]rizkyariesta4@gmail.com, [3]ndengen@gmail.com, [4]meditaruk@gmail.com

Abstrak

Kriptografi merupakan kegiatan pengamanan data baik berupa pesan maupun file dalam bentuk yang lain dari tindak pencurian dan duplikasi dari pihak yang tidak berhak. Pada saat ini khususnya di Publik Khatulistiwa Televisi Bontang, yang merupakan salah satu media televisi yang berada di Kota Bontang ini memiliki permasalahan dalam pengamanan data berita seperti dalam bentuk dokumen, gambar, dan audio. Sebagian besar meningkatnya tindak pencurian serta publikasi ini dilakukan oleh rekan sesama media lain. Sehingga perlu ditingkatkannya pengamanan data oleh pihak internal Publik Khatulistiwa Televisi Bontang. Oleh karena itu, dalam penelitian ini dibangun sebuah aplikasi kriptografi yang dapat mengamankan data berita dengan mengubah isi pesan yang ada dalam data berita. Menggunakan metode kriptografi International Data Encryption Algorithm (IDEA) yang merupakan kriptografi simetris dengan memanfaatkan penggunaan satu kunci saja untuk banyak data dalam proses enkripsi dan dekripsi. Hasil dari pembangunan aplikasi kriptografi data berita pada Publik Khatulistiwa Televisi Bontang diharapkan dapat mengurangi tingkat pencurian dan duplikasi data, serta meningkatkan keamanan data berita dalam internal Publik Khatulistiwa Televisi Bontang maupun eksternal dengan rekan media yang lain.

Kata kunci*— Kriptografi IDEA, Java, Visual Java Netbeans*

1. PENDAHULUAN

Pentingnya menjaga kerahasiaan suatu informasi baik berupa *file* dokumen, audio, maupun video membuat ilmu kriptografi digunakan untuk mengamankan berbagai data, baik data informasi secara umum maupun data penting seperti data berita pada khususnya. Perkembangan data berita menimbulkan berbagai permasalahan seperti penyalahgunaan akses dan penjiplakan yang telah menimbulkan dampak serius terhadap permasalahan legal, sosial, dan ekonomi. Khusunya di Publik Khatulistiwa Televisi Bontang kasus penjiplakan dan pencurian sumber data berita seperti gambar dan video hasil liputan sering terjadi, bahkan kasus penjiplakan dan pencurian data tersebut dilakukan oleh sesama rekan media komunikasi yang lain. Tidak semua data berita diberikan untuk konsumsi masyarakat umum, seperti dokumen penting dari sebuah lembaga pemerintahan sebagai sumber dari berita yang akan ditayangkan. Oleh karena itu berbagai cara dilakukan masyarakat atau bahkan media komunikasi yang lain untuk mendapat informasi yang terdapat pada data berita tersebut. Teknologi baru telah meningkatkan kebutuhan akan keamanan multimedia serta perlindungan hak cipta. Maka, hal ini mengakibatkan kebutuhan keamanan dalam penyimpanan data berita menjadi sangat penting. Sehingga berbagai cara

digunakan untuk melindungi data tersebut agar tidak diketahui oleh orang yang tidak memiliki hak atas informasi yang terdapat pada data berita tersebut.

IDEA (*International Data Encryption Algorithm*) merupakan sebuah algoritma kriptografi simetri yang diciptakan pada awalnya sebagai pengganti *Data Encryption Standard* (DES). Algortima ini merupakan algoritma yang menyediakan keamanan cukup tinggi yang menekankan pada keamanan/kerahasiaan kunci yang digunakan. Pada algoritma kriptogafi dengan algoritma IDEA ini, mampu mengatasi masalah seperti penggunaan algoritma *restirected* yang biasa digunakan oleh sekelompok orang untuk bertukar pesan, mereka membuat suatu algoritma enkripsi tersebut hanya dapat diketahui oleh anggota kelompok itu saja, sehingga ketika ada anggota kelompok yang keluar, kemungkinan algoritma tersebut dapat dibocorkan, maka algoritma *restirected* tersebut harus diganti. Dengan algoritma IDEA, dapat mengatasi masalah seperti pencurian dan penjiplakan data dengan menggunakan kunci, yang algoritmanya tidak lagi dirahasiakan, tetapi kunci yang dirahasiakan. Sehingga apabila ada anggota kelompok yang keluar cukup mengganti kuncinya saja.

Dari beberapa penelitian yang ada, menunjukkan bahwa metode IDEA belum banyak diimplementasikan dalam proses enkripsi dan dekripsi data. Membuat penelitian menggunakan metode IDEA menjadi penelitian baru yang dapat menjadi referensi bagi penelitian-penelitian berikutnya.

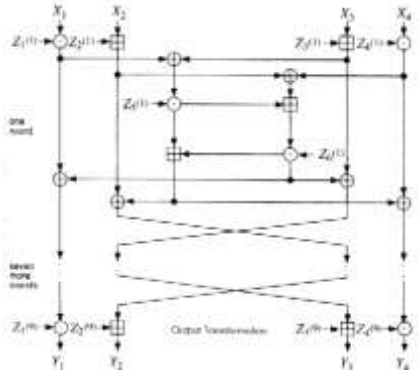
2. METODE PENELITIAN

2.1 Metode International Data Encryption Algorithm (IDEA)

IIDEA menggunakan operasi aljabar yang tidak kompatibel seperti penggunaan opearator XOR, penambahan modulo pada 2^{16} dan perkalian modulo $2^{16} + 1$ (operasi ini menggantikan kotak-S).

2.1.1 Enkripsi IDEA

Blok data 64 bit dibagi menjadi empat subblok 16 bit X_x, X_2, X_3, X_4. Empat subblok ini menjadi masukan bagi iterasi tahap pertama algoritma. Total terdapat 8 iterasi. Pada setiap iterasi, 4 sub-blok di-XOR-kan, ditambahkan, dikalikan dengan yang lain dan dengan 6 sub-kunci 16-bit. Diantara iterasi, sub-blok kedua dan ketiga saling dipertukarkan. Akhirnya 4 sub-blok dikombinasikan dengan 4 sub-kunci dalam transformasi keluaran. Perlu diperhatikan, dalam beberapa literatur mengenai IDEA. Kunci K sering disebut sebagai Z, sebagai mana pula dalam gambar kita di sini.



Gambar 1. Diagram blok IDEA

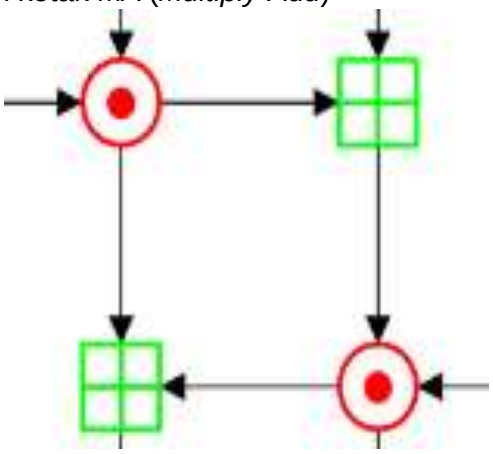
Pada setiap tahapan, urutan berikut ini dikerjakan dimulai dengan mengalikan X_1 dengan sub-kunci pertama Z1(1) atau K1(1). Kemudian menambahkan X_2 dengan sub-kunci kedua. Tambahkan X_3 dengan sub-kunci ketiga. Kalikan X_4 dengan sub-kunci keempat. Keluaran setiap tahapan adalah 4 sub-bok yang merupakan hasil langkah pertukaran 2 subblok dalam (blok 2 dan 3) kecuali pada tahapan terakhir, hal tersebut

adalah input untuk tahapan berikutnya. 4 sub-blok digabungkan kembali untuk menghasilkan ciphertext.

2.1.2 Dekripsi IDEA

Dekripsi dilakukan dengan cara serupa dengan proses enkripsi kecuali pembentukan sub-kuncinya yang berbeda. Blok cipher 64-bit dijadikan masukan, dan keluarannya adalah plaintext. Pembentukan kunci dekripsi berikut berulang 8 kali, dengan menambahkan 6 ke indek kunci dekripsi dan mengurangi 6 dari setiap indek kunci enkripsi. Prinsip Desain IDEA adalah sebagai berikut :

1. Kunci cukup panjang : 128 bit
2. *Confusion* : menggabungkan 3 kelompok operasi yang "tidak kompatibel"(xor, penjumlahan mod 2^{16} dan perkalian mod 2^{17})
3. *Diffusion* : diberikan oleh kotak MA (*Multiply-Add*)



Gambar 2. Pembentukan subkey IDEA

Dimana **(** adalah penjumlahan mod 2^{16} dan " adalah perkalian mod $2^{17.}$ Model ini merupakan jumlah komponen minimum yang diperlukan untuk menghasilkan difusi.

3. HASIL DAN PEMBAHASAN

3.1 Perancangan Data

Berikut contoh perhitungan enkripsi dan dekripsi secara manual dengan variabel Plain Text bernilai "teknikinformatika" dan kunci bernilai "1415015048"

1. Nilai konversi karakter ke binary

Tabel 1**.** Nilai ASCII dari plainteks dan kunci plainteks

Char	Desimal	Binary
T	116	01110100
E	101	01100101
K	107	01101011
N	110	01101110
I	105	01101001
K	107	01101011
I	105	01101001
N	110	01101110
F	102	01100110
O	111	01101111
R	114	01110010
m	109	01101101

A	97	01100001
T	116	01110100
I	105	01101001
K	107	01101011
A	97	01100001
1	49	00110001
4	52	00110100
1	49	00110001
5	53	00110101
0	48	00110000
1	49	00110001
5	53	00110101
0	48	00110000
4	52	00110100
8	56	00111000

2. Kemudian kunci digabungkan.
   01110100011001010110101101101110011010010110101101101001011011100110011001101111011100100110110101100001011101000110100101101011
3. Kunci dipecah menjadi 8 kelompok.
   Ke 1(Putaran 1) = 0111010001100101
   Ke 2(Putaran 1) = 0110101101101110
   Ke 3(Putaran 1) = 0110100101101011
   Ke 4(Putaran 1) = 0110100101101110
   Ke 5(Putaran 1) = 0110011001101111
   Ke 6(Putaran 1) = 0111001001101101
   Ke 1(Putaran 2) = 0110000101110100
   Ke 2(Putaran 2) = 0110100101101011
4. Selanjutnya dilakukan rotasi ke kiri sebanyak 25 karakter
   (01110100011001010110101101101110011010010110101101101001011011100110011001101111011100100110110101100001011101000110100101101011,25)=
   11011100110100101110101101101001011011100110011001101111011100100110101100001011101000110100101101011010110
5. Kemudian menghasilkan kunci baru sebagai berikut :
   a. Kunci kedua
   Ke 3(Putaran 2) = 1101110011010010
   Ke 4(Putaran 2) = 1101011011010010
   Ke 5(Putaran 2) = 1101110011001100
   Ke 6(Putaran 2) = 1101111011100100
   Ke 1(Putaran 3) = 1101101011000010
   Ke 2(Putaran 3) = 1110100011010010
   Ke 3(Putaran 3) = 1101011011101000
   Ke 4(Putaran 3) = 1100101011010110
   Kemudian digabung kembali sehingga menjadi seperti berikut
   11011100110100101110101101101001011011100110011001101111011100100110101100001011101000110100101101011010110
   Langkah selanjutnya digeser ke kiri 25 karakter
   (11011100110100101110101101101001011011100110011001101111011100100110101100001011101000110100101101011010110,25)

=1010010110111001100110011011111011100100110110101100001011101000110100101101011011101000110010101101101101110011010010110101101101

Dilakukan terus proses yang sama hingga putaran ke7 dan mendapat kunci kelima.

b. Kunci kelima

Ke 3(Putaran 6) =	Ke 1(Putaran 7) =
0001011101000110	1001011010110110
Ke 4(Putaran 6) =	Ke 2(Putaran 7) =
1001011010110111	1001011011100110
Ke 5(Putaran 6) =	Ke 3(Putaran 7) =
.0100011001010110	0110011011110111
Ke 6(Putaran 6) =	Ke 4(Putaran 7) =
1011011011100110	0010011011010110

Pada putaran ke 7, hanya 4 pecahan kunci terakhir yang digunakan, sehingga menjadi :

Ke 1 (Transformasi Output) = 1001011010110110 = X_1
Ke 2 (Transformasi Output) = 1001011011100110 = X_2
Ke 3 (Transformasi Output) = 0110011011110111 = X_3
Ke 4 (Transformasi Output) = 0010011011010110 = X_4

3.2 Proses Enkripsi

Tabel 2. Proses Enkripsi

L	Rumus	Nilai Proses	Hasil Proses
1	$(X_1 * K_1) \bmod (2^{16}+1)$	(1001011010110110* 0111010001100101) mod $(2^{16}+1)$	1011001101000000
2	$(X_2 + K_2) \bmod 2^{16}$	(1001011011100110 + 0110101101101110) mod 2^{16}	11001110110100
3	$(X_3 + K_3) \bmod 2^{16}$	(0110011011110111 + 0110100101101011) mod 2^{16}	1001100011101010
4	$(X_4 * K_4) \bmod (2^{16}+1)$	(0010011011010110 * 0110100101101110) mod $(2^{16}+1)$	10000001001101
5	L#1 XOR L#3	1011001101000000 XOR 1001100011101010	10101110101010
6	L#2 XOR L#4	11001110110100 XOR 10000001001101	1001111111001
7	$(L\#5 * K_5) \bmod (2^{16}+1)$	10101110101010 * 0110011001101111 mod $(2^{16}+1)$	1011001010001011
8	$(L\#6 + L\#7) \bmod 2^{16}$	1001111111001 + 1011001010001011 mod 2^{16}	10011100100
9	$(L\#8 * K_6) \bmod (2^{16}+1)$	10011100100 * 0111001001101101 mod $(2^{16}+1)$	110010010111111
10	$(L\#7 + L\#9) \bmod 2^{16}$	1011001010001011 + 110010010111111 mod 2^{16}	1110010100110010
11	L#1 XOR L#9	1011001101000000 XOR 110010010111111	1101011111111111
12	L#3 XOR L#9	1001100011101010 XOR 110010010111111	1111110001010101
13	L#2 XOR L#10	11001110110100 XOR 1110010100110010	1101011010000110
14	L#4 XOR L#10	10000001001101 XOR 1110010100110010	1100010101111111

Kemudian 4 langkah terakhir 11,12,13,14 menjadi nilai X_1, X_2, X_3, X_4 untuk proses selanjutnya. Sehingga diperoleh nilai transformasi dari perhitungan ini :

X_1 = L#11 = 1101011111111111
X_2 = L#12 = 1111110001010101

X_3 = L#13 = 1101011010000110
X_4 = L#14 = 1100010101111111

Nilai-nilai ini kemudian ditransformasikan dengan rumus seperti ini :

Tabel 3. Perhitungan transformasi nilai

Y	Rumus	Nilai Proses	Hasil Proses
1	$(X_1 * K_1) \bmod (2^{16}+1)$	1101011111111111* 1001011010110110 mod $(2^{16}+1)$	1000001011000001
2	$(X_2 + K_2) \bmod 2^{16}$	1111110001010101 + 1001011011100110 mod 2^{16}	1110110010100011
3	$(X_3 + K_3) \bmod 2^{16}$	1101011010000110 + 0110011011110111 mod 2^{16}	101000100111101
4	$(X_4 * K_4) \bmod (2^{16}+1)$	1100010101111111 * 0010011011010110 mod $(2^{16}+1)$	100111010000011

Hasil proses (binary) kemudian diterjemahkan kedalam bentuk karakter ascii sehingga menjadi :

Y_1 = 1000001011000001 = ,Á
Y_2 = 1110110010100011 = ì£
Y_3 = 101000100111101 = ¢=
Y_4 = 100111010000011 = ••

Hasil diatas merupakan hasil akhir dari proses enkripsi sehingga menghasilkan sebuah Cipher text dengan nilai “,Á ì£¢=□□”

3.3 Proses Dekripsi

Pada dasarnya proses yang dilakukan untuk dekripsi ini hamper sama dengan proses enkripsi, terdapat sedikit perbedaan pada 52 buah sub-blok kunci yang digunakan masing-masing merupakan hasil turunan 52 buah sub-blok kunci enkripsi. Berikut bentuk perhitungan manual untuk dekripsi dengan nilai cipher text “,Á ì£¢=□□” dan kunci “1415015048”

1. Nilai konversi karakter ke binary

Tabel 4. Konversi karakter ke ASCII

Char	Desimal	Binary
,	130	10000010
Á	193	11000001
ì	236	11101100
£	163	10100011
¢	162	10100010
=	61	00111101
•	130	10000010
•	130	10000010

2. Kemudian kunci digabung menjadi seperti ini 01101110100011001010110101101101110011010010110101101101001011011100110011001101111011100100110110101100001011101000110100101101

3. Kunci dipecah menjadi 8 kelompok

Ke 1 (Putaran 1)=		Ke 5 (Putaran 1)=	
0110111010001100		1100110011001101	
Ke 2 (Putaran 1)=		Ke 6 (Putaran 1)=	
1010110101101101		1110111001001101	
Ke 3 (Putaran 1)=		Ke 1 (Putaran 2)=	
1100110100101101		1010110000101110	
Ke 4 (Putaran 1)=		Ke 2 (Putaran 2)=	
0110110100101101		1000110100101101	

4. Selanjutnya dilakukan rotasi ke kiri sebanyak 25 karakter
   (01101110100011001010110101101101110011010010110101101101001001011011100110011001100110111011100100110110101100001011101000110100101101011011101.0001100101011010)
   Kemudian menghasilkan kunci baru sebagai berikut
   Kunci Kedua

Ke 3 (Putaran 2)=		Ke 1 (Putaran 3)=	
1101101110011010		1001101101011000	
Ke 4 (Putaran 2)=		Ke 2 (Putaran 3)=	
0101101011011010		0101110100011010	
Ke 5 (Putaran 2)=		Ke 3 (Putaran 3)=	
0101101110011001		0101101011011101	
Ke 6 (Putaran 2)=		Ke 4 (Putaran 3)=	
1001101111011100		0001100101011010	

   Kemudian digabung kembali sehingga menjadi seperti berikut
   1101101110011010010110101101101001011011100110011001101111011100100110110101100001011101000110100101101011011101110010001101011010
   Langkah selanjutnya digeser ke kiri 25 karakter
   (1101101110011010010110101101101001011011100110011001101111011100100110110101100001011101000110100101101011011101110010001101011010, 25)
   =1011010010110111001100110011011110111001001101101011000010111010001101001011010110111001101001011010110111001101001011010110

5. Sampai pada putaran ke 7, hanya 4 pecahan kunci terakhir yang digunakan, sehingga menjadi :
   Ke 1 (Transformasi Output) = 1110011010010110 = X_1
   Ke 2 (Transformasi Output) = 1011011010010110 = X_2
   Ke 3 (Transformasi Output) = 1110011001100110 = X_3
   Ke 4 (Transformasi Output) = 1111011100100110 = X_4

Tabel 5. Perhitungan proses akhir

L	Rumus	Nilai Proses	Hasil Proses
1	$(X_1 * K_1) \bmod (2^{16}+1)$	(1000001011000001* 0110111010001100) mod $(2^{16}+1)$	1000010110011011
2	$(X_2 + K_2) \bmod 2^{16}$	(1110110010100011+ 1010110101101101) mod 2^{16}	1101001000100000
3	$(X_3 + K_3) \bmod 2^{16}$	(1010001000111101 + 1100110100101101) mod 2^{16}	10001111010010
4	$(X_4 * K_4) \bmod (2^{16}+1)$	(1000001010000010 * 0110110100101101) mod $(2^{16}+1)$	1111010111000011

5	L#1 XOR L#3	1000010110011011 XOR 10001111010010	1010011001001001
6	L#2 XOR L#4	110100100010000 XOR 1111010111000011	1001110011010011
7	(L#5 * K_5) mod (2^{16}+1)	1010011001001001 * 1100110011001101 mod (2^{16}+1)	10000011100010
8	(L#6 + L#7) mod 2^{16}	1001110011010011 + 10000011100010 mod 2^{16}	10000110010101
9	(L#8 * K_6) mod (2^{16}+1)	10000110010101 * 1110111001001101 mod (2^{16}+1)	111011101010000
10	(L#7 + L#9) mod 2^{16}	10000011100010 + 111011101010000 mod 2^{16}	10010110111010
11	L#1 XOR L#9	1000010110011011 XOR 111011101010000	1111001011001011
12	L#3 XOR L#9	10001111010010 XOR 111011101010000	101010010000010
13	L#2 XOR L#10	11001110110100 XOR 1110010100110010	1101011010000110
14	L#4 XOR L#10	10000001001101 XOR 1110010100110010	1100010101111111

Kemudian 4 langkah terakhir 11,12,13,14 menjadi nilai X_1, X_2, X_3, X_4 untuk proses selanjutnya. Sehingga diperoleh nilai transformasi dari perhitungan ini :

X_1 = L#11 = 1111001011001011
X_2 = L#12 = 101010010000010
X_3 = L#13 = 1101011010000110
X_4 = L#14 = 1100010101111111

Nilai-nilai ini kemudian ditransformasikan dengan rumus seperti ini :

Tabel 6. Perhitungan Transformasi Nilai

Y	Rumus	Nilai Proses	Hasil Proses
1	(X_1 * K_1) mod (2^{16}+1)	1111001011001011* 0110111010001100 mod (2^{16}+1)	0111010001100101
2	(X_2 + K_2) mod 2^{16}	101010010000010 + 1010110101101101 mod 2^{16}	0110101101101110
3	(X_3 + K_3) mod 2^{16}	1101011010000110 + 1100110100101101 mod 2^{16}	0110100101101011
4	(X_4 * K_4) mod (2^{16}+1)	1100010101111111 * 0110110100101101 mod (2^{16}+1)	0110100101101110

Hasil proses (binary) kemudian diterjemahkan kedalam bentuk karakter ascii sehingga menjadi :

Y_1 = 0111010001100101 = te
Y_2 = 0110101101101110 = kn
Y_3 = 0110100101101011 = ik
Y_4 = 0110100101101110 = in

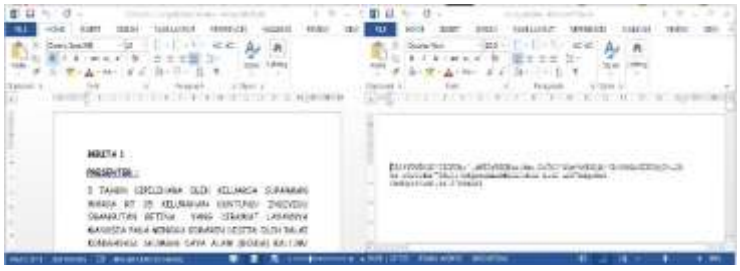
Hasil diatas merupakan hasil akhir dari proses dekripsi sehingga menghasilkan sebuah Cipher text dengan nilai “Õ†7 ó£Ì

3.4 Implementasi Sistem

Program yang dibuat diuji coba dengan mengenkripsi data berupa audio, dokumen dan gambar dengan berbagai variasi format, serta ekstensi *file* yang berbeda.

Gambar 3. Perbedaan audio setelah enkripsi

Gambar 4. Hasil audio sesudah dekripsi



Gambar 5. Perbedaan dokumen sesudah enkripsi

4. KESIMPULAN

Berdasarkan hasil pengujian dan analisis sistem aplikasi kriptografi data berita Publik Khatulistiwa Televisi Bontang, maka dapat disimpulkan bahwa aplikasi ini mampu melakukan proses enkripsi serta dekripsi pada *file* berita yang akan dirahasiakan pesannya, dengan lebih aman karena proses yang sudah terkomputerisasi. Aplikasi kriptografi data berita Publik Khatulistiwa Televisi Bontang ini dapat menjamin keamanan data berita, karena hanya bisa diakses oleh koordinator berita dan bagian produksi.

Selanjutnya dari analisis uji hasil perhitubgan yang dilakukan, waktu yang dibutuhkan untuk melakukan proses enkripsi maupun dekripsi tergantung dari besar nya ukuran *file* yang tersedia. serta tidak terdapat perbedaan ukuran *file* dari proses sebelum enkripsi maupun setelah dekripsi.

5. SARAN

Aplikasi memerlukan GUI (*Graphic User Interface*) yang lebih baik lagi, sehingga memudahkan *user* dalam mengoperasikan aplikasi.

DAFTAR PUSTAKA

[1] Andre Chandra, Willy Sudiarto R, dan Junius Karel T. 2015. “Implementasi Steganografi Dengan Metode End Of File Pada Teks Yang Terenkripsi Menggunakan Block Cipher Rivest Code-6 Ke Dalam Citra”. Jurnal Informatika.

[2] Budiyant, Linda. 2015. “Kriptografi Pada File Dokumen Untuk Keamanan Data Menggunakan Metode Blowfish”. *Skripsi* . Fakultas Matematika dan Ilmu Pengetahuan. Ilmu Komputer. Universitas Mulawarman.

[3] Busran, dan Novernus Ayundha Putra. 2014. “Rekayasa Perangkat Lunak Kriptografi Menggunakan Algoritma RSA Pada Sistem Keamanan File Berbasis Java”. Jurnal TEKNOIF.

[4] Deswintiani Sihotang. 2017. “Perancangan Aplikasi Keamanan Data Text Dengan Metode IDEA dan Kompresi Menggunakan Algoritma Huffman”. Jurnal Majalah Ilmiah INTI Volume : XII.

[5] Dr. Sianipar, Eng RH. 2017. “Kompilasi Proyek KRIPTOGRAFI dengan Visual Basic.Net”. Jakarta. Andi Offset.

[6] Dr. Sianipar, Eng RH. 2017. “Java Untuk Kriptografi”. Jakarta. Andi Offset.

[7] Heri Kurniawan. 2016. “Perancangan Aplikasi Chatting Menggunakan Algoritma Vignere Cipher Sebagai Pengaman Pesan Dalam Jaringan Local Area Network Berbasis Java”. Jurnal Skripsi Universitas Potensi Utama.

[8] Ir. Kurniawan, Yusuf, MT. 2004. “KRIPTOGRAFI Keamanan Internet dan Jaringan Komunikasi”. Bandung. INFORMATIKA.

[9] Jubilee Enterprise. 2015. “Mengenal Java dan Database dengan Netbeans”. Jakarta. PT. Elex Media Komputindo.

[10] Kromodimoeljo, Sentot. 2009. “Teori & Aplikasi Kriptografi”. SPK IT Consuling.

[11] Nugraha, Fajri. 2014. “Kriptografi Pada Citra Digital Menggunakan Metode Hill Cipher”. *Skripsi.* Fakultas Matematika dan Ilmu Pengetahuan Alam. Ilmu Komputer. Universitas Mulawarman.

[12] Rosa A.S dan M. Shalahuddin. 2013. “Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek”. Bandung. INFORMATIKA