

## Aplikasi Penyandian Pesan Teks dengan menggunakan Metode Block Cipher Mode ECB

Francius Manurung

STMIK Budi Darma Medan, Jl. SM.Raja No.338 Sp.Limun Medan, Sumut, Indonesia  
E-Mail : fransfrancius@gmail.com

### ABSTRACT

Encoding of text messages is very necessary to encode a message so that it is not stolen by irresponsible parties. Cryptography is one technique that can be used to carry out security messages, namely by encoding original messages that are understood to be incomprehensible data. Cryptographic techniques are used using the Block Cipher mode ECB method that works by using 64-bit keys and an iterative method consisting of linear steps (such as XOR and key applications). ECB block cipher mode application is to use J2ME as a testing tool. The text message will be entered into Netbeans and tested for the results of encryption and decryption. The design of system testing is done in several stages, namely making use case diagrams, Activity diagrams, Sequence diagrams and design interfaces

Keywords: Cryptography, Applications, Text Message Encoding, ECB Block Cipher mode, J2ME.

### PENDAHULUAN

Kerahasiaan dan keamanan data adalah hal yang sangat penting dalam komunikasi data, baik dengan tujuan keamanan bersama, maupun untuk privasi individu. Para pengguna komputer yang menginginkan agar datanya tidak diketahui oleh pihak-pihak yang tidak berkepentingan selalu berusaha menyasiasi cara mengamankan informasi yang akan dikomunikasikan atau yang akan disimpan. Sehingga perlindungan terhadap kerahasiaan data meningkat, salah satu cara penyandian teks dalam *enkripsi* dan *dekripsi*[1].

Enkripsi merupakan suatu proses perubahan pesan asli menjadi karakter yang tidak dapat dibaca. Ada beberapa algoritma enkripsi yang biasa digunakan seperti *Block Cipher*, *Stream Cipher*, *Data Encryption Standart (DES)*, dan sebagainya. Dimana setiap algoritma memiliki karakteristik tersendiri. Sedangkan proses perubahan kembali hasil *enkripsi* menjadi pesan asli dinamakan *dekripsi*. Untuk merahasiakan data yang sangat penting maka digunakanlah metode kriptografi yang mengenkripsi dan depenelitiankan data.

Salah satu metode yang akan digunakan dalam pembuatan penyandian pesan teks ini adalah metode *Block Cipher mode ECB (Electronic Code Book)*, karena metode ini diimplementasikan pada level *binary digit (bit)*, sehingga pola proses enkripsi tidak dapat terbaca, serta proses *enkripsi* dan *dekripsi* memerlukan waktu yang sangat singkat.

Dalam menjaga keamanan data terdapat sebuah metode pengamanan data yang dikenal dengan nama kriptografi. Kriptografi merupakan salah satu metode pengamanan data yang dapat digunakan untuk menjaga kerahasiaan data, keaslian data serta kerahasiaan pengirim. Penyandian pesan sangat penting untuk dilakukan supaya jika pengirim mengirimkan pesan ke penerima orang lain tidak dapat mengetahui atau merubah data tersebut.

Proses yang dilakukan untuk menyandikan sebuah pesan (yang disebut *plaintext*) mejadi pesan tersembunyi (disebut *ciphertext*) adalah *enkripsi*. *Ciphertext* adalah pesan yang sudah tidak dapat dibaca dengan mudah. Menurut ISO 7498-2, terminologi yang tepat digunakan adalah *encipher*. Proses sebaliknya, untuk mengubah *ciphertext* menjadi *plaintext*, disebut *dekripsi*. Menurut ISO 7498-2, terminologi yang tepat untuk proses ini adalah *dechipper*. Algoritma dari *enkripsi* adalah fungsi-fungsi yang digunakan untuk melakukan *enkripsi* dan *dekripsi*. Algoritma yang digunakan menentukan kekuatan dari *enkripsi*, dan ini biasanya dibuktikan dengan basis matematika. Berdasarkan cara memproses teks (*plaintext*), *Block cipher* bekerja dengan memproses data secara blok, dimana beberapa karakter / data digabungkan menjadi satu blok menghasilkan keluaran satu blok juga.

*Enkripsi* digunakan untuk menyandikan data-data atau informasi sehingga tidak dapat dibaca oleh orang tidak berhak. Pesan yang hendak dikirim dalam bentuk blok-blok besar

(misal 64-bit) dimana blok-blok ini dioperasikan dengan fungsi enkripsi yang sama dan akan menghasilkan informasi rahasia dalam blok-blok yang berukuran sama. Dengan blok *cipher*, blok plaintext yang sama akan dienkripsi menjadi blok ciphertext yang sama bila digunakan kunci yang sama pada kunci sebelumnya.

Agar lebih jelas, terdapat batasan-batasan masalah yang akan dibahas lebih khusus difokuskan pada:

1. Dalam pembuatan aplikasi ini hanya akan membahas mengenai penyandian pesan *text* dengan metode Block Cipher mode ECB (*Electronic Code Book*) dengan panjang *text* 24 karakter.
2. Aplikasi yang berjalan pada komputer dengan spesifikasi yang mendukung Java dengan CLDC 1.1, MIDP 2.1.
3. Aplikasi dibuat dengan menggunakan bahasa pemrograman J2ME, dimana tools yang digunakan adalah Netbeans 6.0.

## LANDASAN TEORI

### 2.1 Aplikasi

Aplikasi berasal dari kata *application* yang artinya penerapan, lamaran, penggunaan. Secara istilah aplikasi adalah program siap pakai yang direka untuk melaksanakan suatu fungsi bagi pengguna atau aplikasi yang lain dapat digunakan oleh sasaran yang dituju. Perangkat lunak aplikasi adalah suatu subkelas perangkat lunak komputer yang memanfaatkan kemampuan komputer langsung untuk melakukan tugas yang diinginkan pengguna. Contoh utama perangkat lunak aplikasi adalah pengolah kata, lembar kerja, dan pemutar media [6].

### 2.2 Kriptografi

Kriptografi (*cryptography*) berasal dari Bahasa Yunani, terdiri dari dua suku kata yaitu "*cryptos*" artinya "*secret*" (rahasia), sedangkan "*graphein*" artinya "*writing*" (tulisan), Jadi, kriptografi berarti "*secret writing*" (tulisan rahasia). Kriptografi adalah ilmu yang mempelajari teknik-teknik matematika yang berhubungan dengan aspek keamanan informasi, seperti kerahasiaan, keabsahan data, integritas data, serta autentikasi data [5]. Secara umum kriptografi adalah ilmu dan seni untuk menjaga kerahasiaan berita (Bruce Schneier – *Applied Cryptography*).

Selain definisi tersebut diatas, terdapat pula definisi yang dikemukakan didalam Kriptografi adalah ilmu yang mempelajari teknik-teknik matematika yang berhubungan dengan aspek keamanan informasi, seperti

kerahasiaan, integritas data, serta otentikasi. (Rinaldi Munir, 2006: 2)

Ada empat tujuan mendasar dari ilmu kriptografi ini yang juga merupakan aspek keamanan informasi yaitu :

1. Kerahasiaan, adalah layanan yang ditujukan untuk menjaga agar pesan tidak dapat dibaca oleh pihak-pihak yang tidak berhak
2. Integritas data, adalah layanan yang menjamin pesan masih asli/utuh atau belum pernah dimanipulasi selama pengiriman
3. Otentikasi, adalah layanan yang berhubungan dengan identifikasi, baik mengidentifikasi kebenaran pihak-pihak yang berkomunikasi (*user authentication* atau *entity authentication*) maupun mengidentifikasi kebenaran sumber pesan (*origin authentication*)
4. Nirpenyangkalan (*non-repudiation*), adalah layanan untuk mencegah entitas yang berkomunikasi melakukan penyangkalan, yaitu pengirim pesan menyangkal melakukan pengiriman atau penerima pesan menyangkal telah menerima pesan. (Rinaldi munir, 2006: 9)

#### 2.2.1 Sejarah Kriptografi

Kriptografi mempunyai sejarah yang panjang. Informasi yang lengkap mengenai sejarah kriptografi dapat ditemukan didalam buku David Kahn yang berjudul *The Codebreakers*. Buku yang tebalnya 1000 halaman ini menulis secara rinci sejarah kriptografi mulai dari penggunaan kriptografi oleh bangsa mesir 4000 tahun yang lalu (berupa *hieroglyph* yang tidak standard pada piramid) hingga penggunaan kriptografi pada abad ke-20. Secara historis ada empat kelompok orang yang berkontribusi terhadap perkembangan kriptografi, dimana mereka menggunakan kriptografi untuk menjamin kerahasiaan dalam komunikasi pesan penting, yaitu kalangan militer (termasuk *intelijen* dan mata-mata) , kalangan diplomatik, penulis buku harian, dan pencinta (*lovers*). Diantara keempat kelompok ini, kalangan militer yang memberikan kontribusi paling penting, karena pengiriman pesan didalam suasana perang membutuhkan teknik enkripsi dan dekripsi yang rumit.

Sejarah kriptografi sebagian besar merupakan sejarah kriptografi klasik, yaitu metode enkripsi yang menggunakan kertas dan pesil atau mungkin dengan bantuan alat mekanik sederhana. Secara umum algoritma kriptografi klasik dikelompokkan menjadi dua kategori, yaitu algoritma transposisi (*transposition cipher*) dan

algoritma substitusi (*substitution cipher*). *Cipher* transposisi mengubah susunan huruf-huruf didalam pesan, sedangkan *cipher* substitusi mengganti setiap huruf atau kelompok huruf dengan sebuah huruf atau kelompok huruf lain.

*Scytale* terdiri dari sebuah kertas panjang dari daun *papyrus* yang dililitkan pada sebuah silinder dari diameter tertentu (diameter silinder menyatakan kunci penyandian). Pesan ditulis secara *horizontal*, baris per baris. Bila pita dilepas, maka huruf-huruf didalamnya telah tersusun secara acak membentuk pesan rahasia. Untuk membaca pesan, penerima pesan harus melilitkan kembali, melilitkan kertas tersebut ke silinder yang diameternya sama dengan diameter silinder pengirim. Sedangkan algoritma substitusi paling awal dan paling sederhana adalah *Caesar cipher*, yang digunakan oleh raja yunani kuno, Julius Caesar. Caranya adalah dengan mengganti setiap karakter didalam *alfabet* dengan karakter yang terletak pada tiga posisi berikutnya didalam susunan *alphabet* [5].



Gambar 1 Pesan ditulis secara horizontal [5]

### 2.2.2 Enkripsi dan Dekripsi

Proses penyandian *plaintexts* menjadi *ciphertexts* disebut Enkripsi (*encryption*) atau *enciphering* (standard nama menurut ISO 7498-2), sedangkan proses mengembalikan *ciphertexts* menjadi *plaintexts* semula dinamakan Dekripsi (*decryption*) atau *deciphering* (standard nama menurut ISO 7498-2).

Enkripsi dan Dekripsi dapat diterapkan baik pada pesan yang dikirim maupun pada pesan tersimpan. Istilah *encryption of data in motion* mengacu pada enkripsi pesan yang ditransmisikan melalui saluran komunikasi, sedangkan istilah *encryption of data at-rest* mengacu pada enkripsi dokumen yang disimpan didalam storage. Contoh *encryption of data in motion* adalah pengiriman nomor PIN dari mesin ATM ke *computer server* di kantor bank pusat. (Rinaldi Munir, 2006: 4)

### 2.3 Algoritma Block Cipher

*Block Cipher* adalah algoritma enkripsi yang akan membagi-bagi *plaintext* yang akan

dikirimkan dengan ukuran tertentu (disebut blok) dengan panjang  $t$ , dan setiap blok dienkripsi dengan menggunakan kunci yang sama. Pada umumnya, *block cipher* memproses *plaintext* dengan blok yang relatif panjang lebih dari 64 bit, untuk mempersulit penggunaan pola-pola serangan yang ada untuk membongkar kunci. Pada cipher blok, rangkaian bit-bit *plaintexts* dibagi menjadi blok-blok bit dengan panjang sama, biasanya 64 bit (tapi adakalanya lebih). Algoritma enkripsi menghasilkan blok *ciphertexts* yang pada kebanyakan sistem kriptografi simetri berukuran sama dengan blok *plaintexts* [5]. Dengan blok cipher, blok *plaintexts* yang sama akan dienkripsi menjadi blok *ciphertexts* yang sama bila digunakan kunci yang sama pula. Ini berbeda dengan cipher aliran dimana bit-bit *plaintexts* yang sama akan dienkripsi menjadi bit-bit *ciphertexts* yang berbeda setiap kali dienkripsi.

Misalkan blok *plaintexts* ( $P$ ) yang berukuran  $m$  bit dinyatakan sebagai vektor

$$P = (P_1, P_2, P_3, \dots, P_m)$$

Yang dalam hal ini  $P_i$  adalah 0 atau 1 untuk  $i = 1, 2, \dots, n$ , dan blok *ciphertexts* ( $C$ ) adalah

$$C = (C_1, C_2, \dots, C_n)$$

Yang dalam hal ini  $C_i$  adalah 0 atau 1 untuk  $i = 1, 2, \dots, n$ . Bila *plaintexts* dibagi menjadi  $m$  buah blok, barisan blok – blok *plaintexts* dinyatakan sebagai

$$(P_1, P_2, \dots, P_m)$$

Untuk setiap blok *plaintexts*  $P_i$  bit-bit penyusunannya dapat dinyatakan sebagai vektor

$$P_i = (P_{i1}, P_{i2}, \dots, P_{in})$$

Enpenelitian dan dekripsi dengan kunci  $K$  dinyatakan berturut-turut dengan persamaan

$$E_k(P) = C \quad (\text{enkripsi})$$

Dan

$$D_k(C) = P \quad (\text{dekripsi})$$

Fungsi  $E$  haruslah fungsi yang berkoresponden satu-ke-satu, sehingga

$$E^{-1} = D$$

Untuk menambah kehandalan algoritma ini, dikembangkan pula tipe proses enkripsi, yaitu:

#### a. ECB (*Electronic Code Book*)

Model operasi EBC adalah implementasi DES yang paling sederhana. Setiap 64 bit *plaintext* dikodekan sendiri-sendiri menjadi *ciphertext* dengan kunci yang sama. Jadi, jika *plaintext* terdiri dari 128 bit, maka setiap 64 bit akan dikodekan secara terpisah. Model semacam ini biasa diimplementasikan pada aplikasi yang terkait dengan transmisi data tunggal atau data yang tidak terlalu besar seperti transmisi kunci. Jadi, jika anda ingin

mentransmisikan kunci untuk enkripsi atau dekripsi, anda dapat menggunakan model operasi ECB ini. Secara matematis, enkripsi dan dekripsi dengan mode ECB dinyatakan sebagai:

$$C_i = E_k(P_i)$$

dan dekripsi sebagai

$$P_i = D_k(C_i)$$

Ket: P : Plaintext

C : Ciphertext

K : Kunci

E : Enkripsi

D : Depenelitian

Contoh 3. Tinjau kembali plaintexts (dalam biner) pada Contoh 1:

10100010001110101001

Bagi plaintexts menjadi blok-blok yang berukuran 4 bit:

1010 0010 0011 1010 1001

atau dalam notasi HEX adalah A23A9[5]

## PEMBAHASAN

### 3.1 Analisa

Masalah utama pada penelitian ini adalah aplikasi kriptografi yang menggunakan algoritma *Block cipher* mode ECB. Penyandian pesan teks sangat penting untuk dilakukan supaya jika pengirim mengirimkan pesan ke penerima orang lain tidak dapat mengetahui atau merubah data tersebut. Kriptografi selalu terdiri dari dua macam yaitu *enkripsi* dan *depenelitian*. Teknik untuk menyandian *plaintext* menjadi *ciphertext* disebut enkripsi, sedangkan proses mengembalikan *ciphertext* menjadi *plaintext* seperti semula dinamakan *depenelitian*. Ditinjau sementara diperoleh permasalahan yaitu dari segi penyandian pesan teks.

Aplikasi penyandian pesan teks menggunakan bahasa pemrograman Java dimana *platform* yang digunakan adalah J2ME (*Java 2 Micro Edition*). J2ME merupakan edisi khusus dari Java dan subset dari edisi J2SE. Edisi ini untuk pemrograman dengan peralatan-peralatan kecil atau terbatas seperti, PDA, *Handphone*, *pager* dan lain-lain. Oleh karena itu penyandian pesan teks yang akan dirancang hanya dapat dijalankan pada perangkat *mobile* yang memiliki fasilitas yang mendukung atau *fitur-fitur* yang mendukung Java. Aplikasi penyandian pesan teks yang akan dirancang menggunakan Algoritma *Block Cipher* mode ECB (*Elektronik Code Book*) sebagai solusi dalam *Enkripsi* dan *Dekripsi* penyandian pesan teks.

### 3.2 Penerapan Algoritma Block Cipher mode ECB ( *Electronic Code Book* )

*Enkripsi* dan *dekripsi* yang sifatnya acak ini sangat cocok diimplementasikan dengan algoritma *block cipher* mode ECB ( *Elektronik Code Book* ), dengan syarat setiap record terdiri dari sejumlah banyak diskrit yang sama banyaknya. Mode ECB cocok untuk mengenkripsi *file* yang diakses secara acak karena tiap blok *plaintext* dienkripsi secara independen. Adapun proses cara kerja dari ECB dalam penyandian pesan teks pada *enkripsi* dan *depenelitian* adalah sebagai berikut:

1. Konversikan setiap karakter *plaintext* maupun kunci ke biner.
2. Kelompokkan seluruh biner *plaintext* dimana jumlah bit setiap kelompok disesuaikan dengan ketentuan soal.
3. Jumlah bit kunci harus sama dengan jumlah bit setiap kelompok biner *plaintext / ciphertext*.
4. Lakukan operasi XOR antara kelompok-kelompok *plaintext* dengan kunci.
5. Konversikan hasil XOR tersebut ke heksadesimal.

Dibawah ini adalah kasus perhitungan dari algoritma *Block cipher* mode ECB ( *Elektronik Code Book* ) dengan proses *Enkripsi* adalah sebagai berikut:

*Plaintext* :  
 FRANCIUS\_MANURUNG\_AMPERA  
 Key : CIUS

Jmlh bit setiap kelompok : 32

*Enkripsi* dan *dekripsi* dengan operasi XOR.

F=70 = 01000110  
 R=82 = 01010010  
 A=65 = 01000001  
 N=78 = 01001110

P1

C= 67 = 01000011  
 I = 73 = 01001001  
 U= 85 = 01010101  
 S= 83 = 01010011

P2

\_ = 95 = 01011111  
 M= 77 = 01001101  
 A= 65 = 01000001  
 N=78 = 01001110

P3

U= 85 = 01010101  
 R= 82 = 01010010  
 U= 85 = 01010101  
 N=78 = 01001110

P4

G=71 = 01000111  
 \_=95 = 01011111  
 A=65 = 01000001

P5

M=77 = 01001101

P=80 = 01010000  
 E=69 = 01000101  
 R=82 = 01010010  
 A=65 = 01000001

Kunci : CIUS

C= 67 = 01000011  
 I = 73 = 01001001  
 U= 85 = 01010101  
 S= 83 = 01010011

Bit *plaintext* seluruhnya :

$$P_1 = P_1 \oplus K$$

$$= 01000110 \ 01010010$$

$$01000001 \ 01001110$$

$$01000011 \ 01001001$$


---


$$01010101 \ 01010011 \oplus$$

Hasil XOR = 00000101 00011011  
 00010100 00011101

5 1B  
 14 1D  
 Notasi Hexadesimal = 5 1B 14 1D

$$P_2 = P_2 \oplus K$$

$$= 01000011 \ 01001001$$

$$01010101 \ 01010011$$

$$01000011 \ 01001001$$


---


$$01010101 \ 01010011 \oplus$$

Hasil XOR = 00000000 00000000  
 00000000 00000000

0 0  
 0 0  
 Notasi Hexadesimal = 0 0 0 0

$$P_3 = P_3 \oplus K$$

$$= 01011111 \ 01001101$$

$$01000001 \ 01001110$$

$$01000011 \ 01001001$$


---


$$01010101 \ 01010011 \oplus$$

Hasil XOR = 00011100 00000100  
 00010100 00011101

1C 4  
 14 1D  
 Notasi Hexadesimal = 1C 4 14 1D

$$P_4 = P_4 \oplus K$$

$$= 01010101 \ 01010010$$

$$01010101 \ 01001110$$

$$01000011 \ 01001001$$


---


$$01010101 \ 01010011 \oplus$$

Hasil XOR = 00010110 00011011  
 00000000 00011101

16 1B  
 0 1D  
 Notasi Hexadesimal = 16 1B 0 1D

$$P_5 = P_5 \oplus K$$

$$= 01000111 \ 01011111$$

$$01000001 \ 01001101$$

$$01000011 \ 01001001$$

$$01010101 \ 01010011 \oplus$$


---

Hasil XOR = 00000100 00010110  
 00010100 00011110

4 16  
 14 1E  
 Notasi Hexadesimal = 4 16 14 1E

$$P_6 = P_6 \oplus K$$

$$= 01010000 \ 01000101$$

$$01010010 \ 01000001$$

$$01000011 \ 01001001$$


---


$$01010101 \ 01010011 \oplus$$

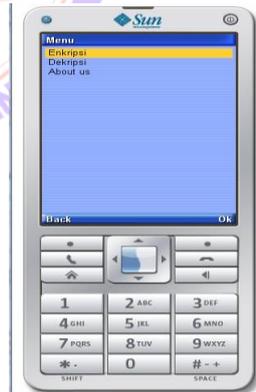
Hasil XOR = 00010011 00001100  
 00000111 00010010

13 C  
 7 12  
 Notasi Hexadesimal = 13 C 7 12

Jadi *Ciphertext* seluruhnya adalah : 5 1B 14 1D, 0 0 0 0, 1C 4 14 1D, 16 1B 0 1D, 4 16 14 1E, 13 C 7 12, Hexadesimal.

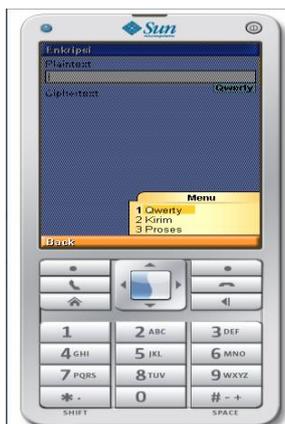
### IMPLEMENTASI

Menu utama merupakan tampilan *list* yang memuat beberapa elemen yaitu menu, *enkripsi*, *dekripsi*, dan *about*. pengaturan, petunjuk, *ticker*, *back* [4]. Penambahan ikon pada *list* ini dimaksudkan menjadikan tampilan menu utama agar menjadi lebih menarik. Tampilan menu utama dapat dilihat pada gambar berikut:



Gambar 2 Tampilan Menu Utama

Di dalam menu *enkripsi* terdapat beberapa tampilan atau sub menu seperti Qwerty, Kirim, dan Proses. Dimana Qwerty berfungsi untuk menampilkan pesan *enkripsi*. Tampilan menu dapat dilihat pada gambar berikut:



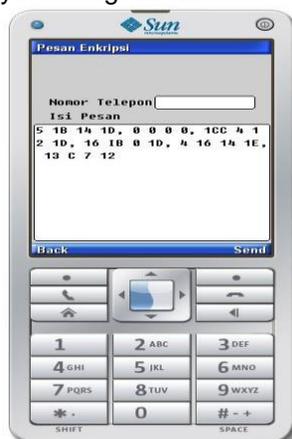
Gambar 3 Menu Enkripsi

Jika pengguna ingin mengenkrip pesan harus menginputkan *plaintext* yang mau *dienkripsi*. tampilannya sebagai berikut:



Gambar 4 Tampilan Pesan Enkripsi

Jika pengguna sudah mengenkrip pesan dan menekan tombol lanjutkan / kirim, maka pengguna akan diminta memasukkan nomer tujuan pesan, tampilannya sebagai berikut :



Gambar 5 Hasil Pesan Terenkripsi

Jika pengguna ingin mendekripsi pesan harus menginputkan *ciphertext* yang mau *didekripsi*. Tampilannya sebagai berikut.



Gambar 6 Form Dekripsi

## KESIMPULAN

Kesimpulan yang didapat dari penulisan penelitian ini adalah sebagai berikut:

1. Metode *Block Chiper* dalam penyediaan pesan teks untuk *enkripsi* dan *dekripsi* pada pesan *text* harus mengkonversikan setiap karakter *plaintext* maupun kunci ke biner. Kelompokkan seluruh biner *plaintext* dimana jumlah bit setiap kelompok sesuaikan dengan ketentuan soal, setelah itu jumlah bit kunci harus sama dengan jumlah bit setiap kelompok biner *plaintext* / *chiptext*. Lakukan operasi XOR antara kelompok-kelompok *plaintext* dengan kunci dan Konversikan hasil XOR tersebut ke heksadesimal.
2. Dalam penyediaan pesan *text* dengan J2ME (*Java 2 Micro Edition*) adalah dengan menggunakan perangkat *mobile* yang memiliki fasilitas yang mendukung atau *fitur-fitur* yang mendukung Java. Aplikasi penyediaan pesan teks yang akan dirancang menggunakan Algoritma *Block Chiper* mode ECB (*Elektronik Code Book*) sebagai solusi dalam *Enkripsi* dan *Dekripsi* penyediaan pesan teks.

## DAFTAR PUSTAKA

- [1]. Adi Nugroho, 2010. "Rekayasa Perangkat Lunak Berorientasi Objek dengan Metode USDP". Penerbit Andi Yogyakarta.
- [2]. Eko Priyo Utomo, 2009. "Java Virtual Machine". Penerbit Yrama Widya Bandung.
- [3]. Kadir, Abdul, 2013. "Pengenalan Algoritma Pendekatan Secara Visual dan Interaktif Menggunakan Raptor". Penerbit Andi Yogyakarta.
- [4]. M. Salahudin dan Rosa A.S, "Pemrograman J2ME Belajar Cepat Pemrograman Perangkat Komunikasi

- Mobile, 2008". Penerbit Informatika Bandung.
- [5]. Rinaldi Munir (2006). "Kriptografi Untuk Keamanan Jaringan dan Implementasinya Dalam Bahasa Java". Penerbit Andi Yogyakarta.
- [6]. <http://pelita-informatika.com/berkas/jurnal/417.pdf> tanggal akses 1 agustus 2013.

