ISSN : 2252-3839 (Print)
ISSN : 2549-2403 (On Line)
DOI : 10.28989/compiler.v8i2.471
http://ejournals.stta.ac.id/index.php/compiler/

COMBINATION OF XOR BINARY ALGORITHM AND STEGANOGRAPHY USING LEAST SIGNIFICANT BIT (LSB) METHOD FOR DATA SECURITY

Yulius Nahak Tetik[1)], Friden Elefri Neno[2)], Dony Ariyus[3)]
[1,2,3] Universitas AMIKOM Yogyakarta Magister Teknik Informatika
Jl Ring Road Utara, Condongcatur, Sleman, Yogyakarta 55281
Email: [1]yuliusteti@gmail.com,[2]nenofriden.e@gmail.com, [3]dony.a@amikom.ac.id

Abstract

Leakage of data or information that is owned by both personal and organizational nature is caused due to negligence of humans or users of the system itself or by hackers who arbitrarily dismantle the system used by users and disseminate data or information. Cryptography is one of the sciences or arts that studies about how to secure data or information from irresponsible parties who want to destroy important data in the form of text and image files. This research, data security technique uses two algorithms, namely steganography to enter text in image media using the LSB (Least Significant Bit) and Binary XOR methods to convert messages into binary with XOR keywords and generate message pixel values from 8 image bits with LSB (The least important bit). Data or text that has been inserted in the next image will be sent to the recipient, and to view the original data, the recipient of the message must decrypt the data with the same key as during the encryption process and image insertion and retrieve the LSB (Least Significant Bit) value, from the image encrypted. Based on the results and tests carried out, the process of encrypting data and inserting messages in pictures can minimize data or information that will be delivered or sent to the recipient.

Keywords: *XOR, LSB, Cryptography, Steganography*

1. Latar Belakang Masalah

Steganografi dalam bahasa Yunani berarti 'steganos' berarti 'tersembunyi atau terselubung dan ' *grephien* ' berarti 'menulis atau menggambar', umum itu berarti 'tulisan tertutup [1]. Tujuan utama dari Steganografi adalah menyembunyikan informasi dalam suatu media (gambar / teks) yang membantu dalam menyembunyikan pentingnya rahasia pesan. Komponen steganografi meliputi pesan sampul (steganos), pesan rahasia (menulis) dan kunci rahasia.

Alasan paling umum adalah bahwa, penyusup dapat memperoleh akses ilegal terhadap informasi dan dapat menyalahgunakannya informasi. Ini tidak hanya terbatas pada informasi atau komunikasi, tetapi juga berlaku pada jaringan komputer karena internet adalah satu-satunya media untuk bertukar informasi. Alasan utama untuk menyediakan keamanan adalah untuk mempertahankan kerahasiaan, integritas, ketersediaan, dan juga untuk menghentikan penggunaan informasi secara ilegal. Cara paling umum untuk menghentikan ini adalah untuk menerapkan Steganografi dan kriptografi. *Image Steganography* adalah area penting di bidang Data Keamanan. Sebagai tuntutan keamanan dan privasi semakin meningkat, kebutuhan untuk menyembunyikan informasi rahasia sangat penting.

Untuk menjaga kerahasian data, data dikirim dalam bentuk plaintext kedalam kata kunci privasi (*ciphertext*) yang tidak dikenali oleh yang berkepentingan. Setelah data sampai kepada penggguna pesan rahasia (*ciphertext*) dikonversikan ke bentuk semula (*plaintext*). Pada penelitian ini kombinasi dua metode XOR binary dan LSB *(Least*

Significant Bit) ,yaitu plainteks pesan atau teks dengan kunci yang dikonversi ke biner menghasilkan cipherteks XOR dengan LSB *(Least Significant Bit)* penyimpanan plainteks pada gambar menggunakan nilai bit 24 atau 8 pada penyimpanan citra gambar.

Beberapa penelitian terlebih dahulu yang dilakukan menjadi referensi penelitian adalah: Penelitian yang dilakukan oleh [2] tentang pendekatan dalam mengamankan data dengan dengan teknik modifikasi terhadap algoritma XOR. tujuan penelitian ini adalah implementasi metode enkripsi XOR dalam enkripsi data yang dipecah menjadi blok-blok untuk penghapusan pola.

Penelitian tentang steganografi dengan metode *Least Significant Bit* (LSB) tujuan penelitian adalah uji coba metode LSB (*Least Significant Bit)* pada penyisipan dan ekstrasi pesan namun metode yang digunakan masih sederhana karena metode yang digunakan tanpa mengkombinasikan dengan metode lain [3].

Penelitian selanjutnya yaitu kriptografi data sederhana dengan Metode *Exlusive-OR (*XOR*)* penelitian ini menguraikan langkah-langkah kriptografi dengan XOR yang diimplementasi ke teknik kriptografi modern dengan kunci simetris metode XOR pada proses enkripsi dan dekripsi kemudian melakukan konversi pesan dan kata kunci ke bilangan biner yang menghasilkan plaintext yang di konversi karakter berdasarkan kode ASCII [4].

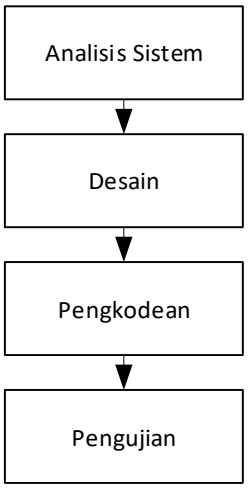
Penelitian tentang peningkatan keamanan data berbasis Eksklusif OR (XOR) untuk lingkungan *Cloud* yang bertujuan untuk analisis secara mendalam algoritma keamanan, algoritma enkripsi berbasis Exclusive OR (XOR) untuk meningkatkan keamanan data [5]

Penelitian tentang keamanan data dengan metode kriptografi XOR. Pada penelitian tersebut bertujuan untuk mengamankan data dengan XOR dengan Teknik penyandian sederhana dalam melakukan enkripsi dan dekripsi data [6].

Penelitian berikutnya tentang modifikasi algoritma *vigenere cipher* menggunakan metode *catalan number* dan *double columnar transposition* tujuan penelitian ini adalah utuk keamanan dukumen data dengan vigenere cipher y menggunakan kunci simetris simetris yang mensubstitusikan plaintext dan dan kata kunci yang digunakan [7]

2. Metodologi Penelitian

Tahapan-tahapan yang digunakan dalam penelitian ini adalah seperti gambar 1.



Gambar 1. Tahapan penelitian

a. Analisis sistem

Proses pengumpulan data dan analysis yang diperlukan dalam pembuatan sistem

b. Desain
   Proses perancangan dan desain antar muka
c. Pengkodean
   Desain yang ditranslasikan ke dalam komputer berupa *source code* yang dieksekusi sesuai dengan perancangan yang dibuat
d. Pengujian
   Fokus pada perangkat lunak, hal ini untuk memastikan keluaran yang diinginkan untuk mengukur algortima yang digunakan

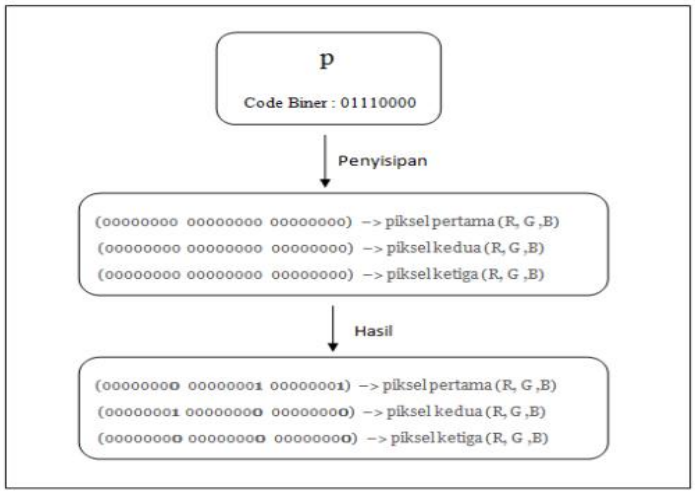
2.1 Kriptografi

Kriptografi adalah ilmu yang mempelajari teknik penyembunyian pesan yang berkaitan dengan keamanan data yang menjamin integritas data, kerahasian dan otentikasi. Pada kriptografi akan menjelaskan tentang cara enkripsi data dari *plaintext* mejadi *chipertext. Plaintext* adalah data dalam keadaan normal yang dibaca oleh manusia sebaliknya *chipertext* adalah kebalikan dari *plaintext* dimana data akan tampak seperti simbol atau karakter tertentu yang hanya dapat dibaca oleh mesin komputer sehingga data atau pesan tersebut akan aman pada saat kirim ke penerima pesan atau data [8]. Kriptografi yang berkaitan dengan aspek keamanan adalah sebagai berikut[9] :

a. *Authentication*
   Penerima dapat menerima informasi dalam bentuk pesan asli dari orang yang mengirim informasi
b. *Integrity*
   Pesan yang dikirim dalam bentuk asli melalui jaringan dan pesan tidak dapat diketahui oleh yang tidak berkepentingan
c. *Non-repudiation*
   Berkaitan dengan pengirim pesan kemudian pengirim tidak menyangkal dengan informasi pesan yang dikirim
d. *Confidentiality*
   Menjaga kerahasiaan informasi data sehingga orang yang tidak berkepentingan tidak bisa mengakses informasi data yang dikirim

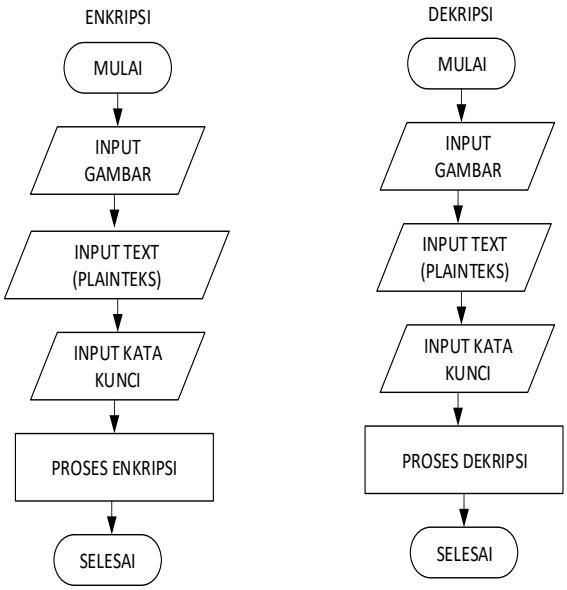
2.2 LSB (Least Significant Bit)

LSB (Least Significant Bit) adalah bilangan biner berbasis 2 (dua) memiliki nilai paling kecil yang tidak berarti yang letaknya pada bagian kanan. Proses metode LSB *(Least Significant Bit),* yaitu teks atau pesan dikonversi ke bilangan biner. Contoh penyisipan huruf “p” kedalam gambar *grayscale* yang dipresentasikan ke biner seperti pada contoh gambar 2. [10].



Gambar 2. Penyisipan huruf “p” kedalam gambar

2.3 Prinsip Kerja Sistem



Gambar 3. *Flowchart* penyisipan teks pada gambar

Berdasarkan *flowchart* pada gambar 3 diatas dapat dijelaskan enkripsi adalah teks asli (*Plaintext*) dan kata kunci dikonversi ke bilangan biner untuk menghasilkan XOR Binary yang disisipkan kedalam bit-bit gambar melalui segmen pixel yang terdiri dari 8-bit ke LSB (*Least Significant Bit)*, kemudian *flowchart* dekripsi untuk membuka teks yang di enkripsi dan konversi menjadi karakter sehingga teks asli dapat dibaca.

Tahapan-tahapan Proses kriptografi XOR [6].

a. Pesan disandikan ke kode ASCII dan di konversikan kedalam bilangan biner
b. Panjang kata kunci sama dengan Panjang pesan
c. Enkripsi kata kunci dan pesan dilakukan dalam XOR

E(i)=P⊕K

Ket:

E=Enkripsi

P=Plaintext

K=Key

d. Pesan yang telah disandikan akan dikembalikan ke kode ASCII
e. Proses Deskripsi pesan dan kata kunci dilakukan operasi XOR

D = C⊕K

Ket:

D = Dekripsi

C = Cipher

K = Key

3. Hasil dan Pembahasan

Untuk perhitungan bilangan *binary* dengan operator XOR dihitung secara matematis adalah ⊕ seperti pada tabel 1.

Tabel 1. Operasi XOR

A	B	A⊕B
1	1	0
0	1	1
0	0	0
1	0	1

Berikut ini adalah perhitungan menggunakan algoritme *Binary* XOR Yang dilakukan proses enkripsi dan dekripsi pada *plaintext* "SAYA" dan kunci "KAMU". Plainteks dan kunci harus diubah menjadi bilangan biner berdasarkan tabel ASCII seperti pada tabel 2.

Tabel 2. Konversi plaintext dan Kunci Menjadi Bilangan Biner Berdasarkan Tabel ASCII

PLAINTEKS	S	A	Y	A
BINER	01010011	01000001	01011001	01000001
KUNCI	K	A	M	U
BINER	01001011	01000001	01001101	01010101

Setelah dilakukan proses konversi, langkah selanjutnya adalah proses enkripsi plainteks terhadap kunci dengan XOR seperti pada tabel 3.

Tabel 3. Proses enkripsi plainteks terhadap kunci dengan XOR

PLAINTEKS	01010011	01000001	01011001	01000001
KUNCI	01001011	01000001	01001101	01010101
XOR	00011000	00000000	00010100	00010100

Pesan "SAYA" setelah dienkripsikan dengan kunci "KAMU" maka akan didapat XOR : **00011000 00000000 00010100 00010100.** XOR tersebut akan disisipkan pada file gambar melalui bit gambar 24 atau 8 bit yang disisipkan pada setiap bit tersebut dari plainteks yang akan menggeserkan posisi LSB (*Least Significant Bit)* dari pixel-pixel gambar.

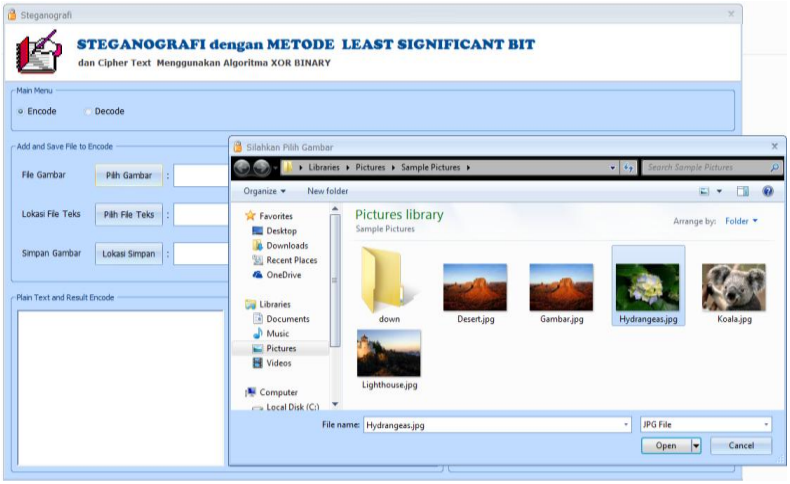
01001011 00101111 00110011 01100010 00101011 00100110
00010110 01001000 01111000 00110010 00100010 01100010

00101010 00100110 00010110 00010110 01001001 00001000
01111001 00100011 00100110 00111001 01111001 00001000
00100011 00100110 01011001 00010000 01100101 00001101
00100111 00100111

01001010 00101110 00110010 01100011 00101011 00100110
00010110 01001000 01111000 00110010 00100010 01100010
00101010 00100110 00010110 00010110 **01001000 01111000**
00001000 **00100011** 00100110 00111001 **01111000** 00001000
00100010 00100110 **01011000 00010001 01100100** 00001101
00100110 01001110

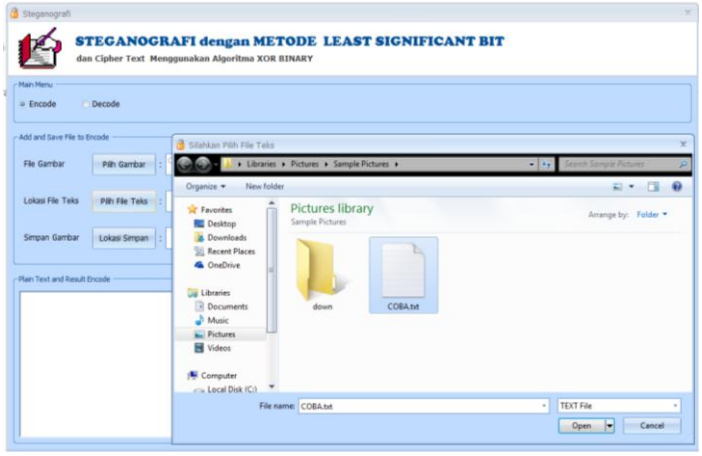
Keterangan: Pesan SAYA direpresentasikan ke bilangan biner pada bit yang dicetak tebal adalah perubahan nilai bit dimana file gambar yang disimpan *plaintext* secara kasat mata manusia tidak membedakan warna gambar yang disimpan dengan gambar asli.

a. *Input* file gambar, teks pesan dan simpan gambar



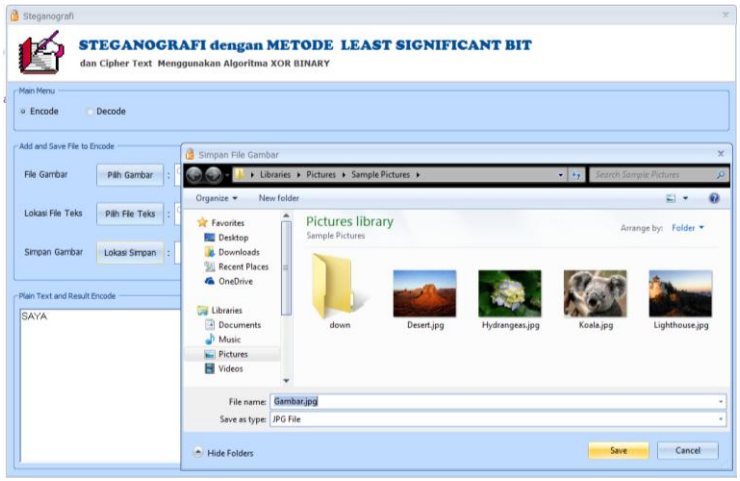
Gambar 4. Penyisipan file gambar

Pada gambar 4 diatas merupakan proses penyisipan gambar dengan cara *klik* pada tombol file gambar, pada kotak dialog yang ditampilkan pilih file gambar dengan format JPG kemudian klik pada tombol tombol *open*, selanjutnya adalah penyisipan file pesan teks yang akan sisipkan pada gambar dengan *klik* pada tombol pilih file teks kemudian pilih dan klik tombol *open* seperti yang ditampilkan pada gambar 5.



Gambar 5. Penyisipan teks pesan

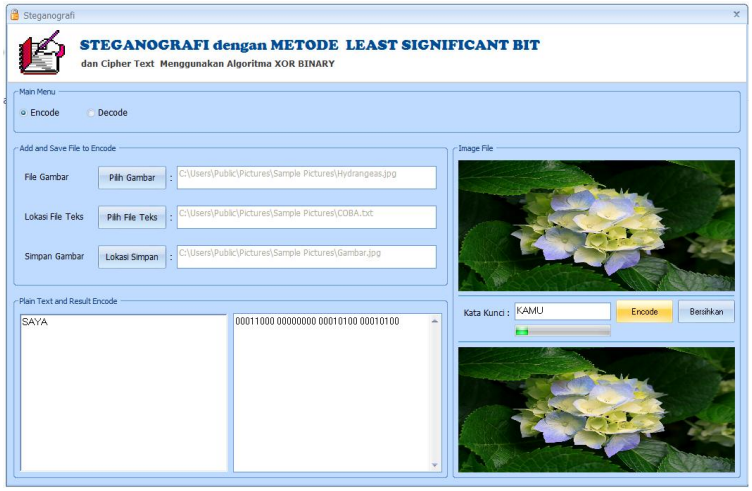
Proses selanjutnya adalah menentukan lokasi atau *directory* dimana file gambar yang akan disisipi pesan tersebut disimpan yaitu dengan pilih tombol lokasi simpan dan pada kotak dialog simpan file gambar, masukan nama file gambar kemudian *klik* pada tombol *save* seperti pada gambar 6.



Gambar 6.

b. Proses enkripsi dan penyisipan pesan teks pada gambar

Setelah proses input *file* yang telah dijelaskan pada poin diatas, selanjutnya adalah proses enkripsi dan penyisipan pesan teks pada gambar dengan kata kunci KAMU kemudian pilih dan *klik* pada tombol *Encode* berikut ini hasil enkripsi dan penyisipan pesan teks pada gambar seperti pada contoh gambar 7.



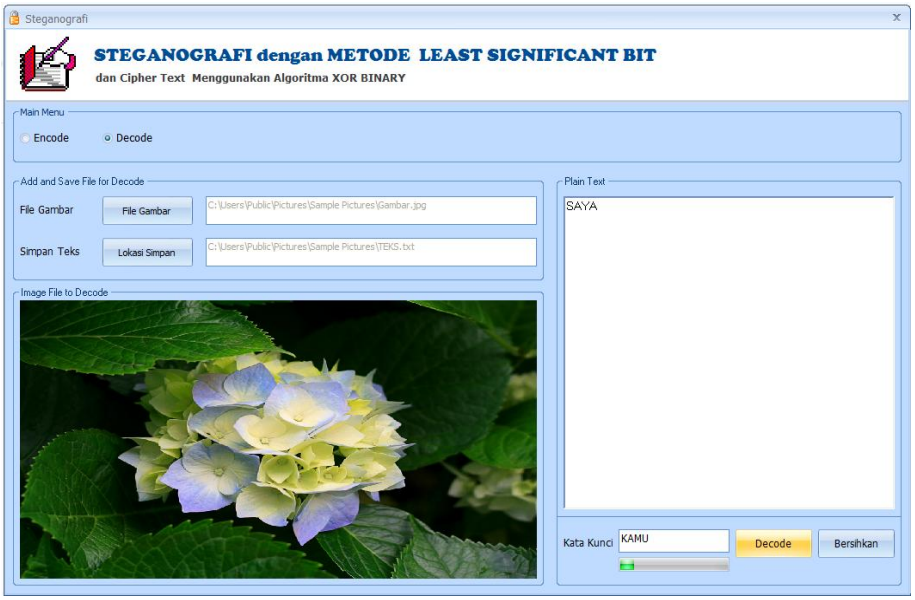
Gambar 7. Hasil enkripsi dan penyisipan pesan pada gambar.

c. Proses pengambilan pesan dari gambar

Proses pengambilan pesan pada gambar dilakukan dengan beberapa langka diantaranya :

1. Klik ada option *button decode*
2. *Input file* gambar yang telah di disisipi pesan teks dengan cara *klik* pada tombol *file* gambar. Pada kotak dialog yang ditampilkan, pilih *file* gambar kemudian *klik* pada tombol open.
3. Menentukan lokasi dimana *file* pesan teks akan disimpan dengan cara klik pada tombol lokasi simpan. Pada kotak dialog yang ditampilkan, masukan nama *file* dan klik tombol *save*.
4. Masukan kata kunci yang telah digunakan sebelumnya pada proses penyisipan pesan teks pada gambar, kemudian klik tombol *Decode.*

Berikut ini gambar hasil *decode* atau pengambilan pesan seperti yang ditampilkan pada gambar 8.



Gambar 8. Hasil pengambilan pesan teks pada gambar

5. Kesimpulan

Dari hasil pengujian dan pembahasan pada aplikasi yang telah dilakukan maka dapat diambil beberapa kesimpulan sebagai berikut :

1. Aplikasi yang telah dibangun dapat digunakan untuk melakukan enkripsi dan dekripsi pada teks pesan dengan metode *Least Significant Bit* (LSB) dapat digunakan untuk steganografi pada *file* gambar.
2. Proses enkripsi dan dekripsi dilakukan dengan mengonversi pesan dan kata kunci dari XOR ke biner
3. Proses steganografi dengan metode metode *Least Significant Bit* (LSB) dengan cara mengganti posisi bit paling tidak berarti (bit paling kanan) pada gambar dengan pesan yang telah dienkripsikan dalam bentuk bilangan biner.

Daftar Pustaka

[1] Darwis, D. (2016). Implemetasi Teknik Steganografi Least Significant Bit (LSB) dan Kompresi untuk Pengamanan Data Pengiriman Surat Elektronik. *TEKNOINFO, 10*(2), 1-7.

[2] T, O. E., & A, A. O. (2014). An Approach to Improve Data Security using Modified XOR Encryption Algorithm. *IJCRCE, 1*(2), 1-9.

[3] Sitorus, M. (2015). Teknik Steganografi dengan Metode Least Significant Bit (LSB). *Jurnal Ilmiah Fakultas Teknik LIMIT'S, 11*(2), 54-59.

[4] Suhardi. (2016). Aplikasi Kriptografi Sederhana dengan Metode Excluve-OR (XOR). *Teknovasi, 3*(2), 23-31.

[5] Kumar, P., Jangra, S., & Singh, S. (2017). Exclusive or (XOR) Based Enhanced Data Security Algorithm For Cloud Environment. *IJARCS, 8*(5), 1482-1485.

[6] Lubis, J. H. (2018). Implementasi Keamanan Data dengan Metode Kriptografi XOR. *JSIK, 2*(2), 1-4.

[7] Pratama, G. M., & Tamatjita, E. N. (2015). Modifikasi Algoritma Vigenere Cipher menggunakan Metode Catalan Number dan Double Columnar Transportation. *Compiler, 4*(1), 31-40.

[8] Muttaqin, S. H., & Dewi, N. (2014). Penerapan Sistem Keamanan menggunakan Cryptography pada Aplikasi Chatting dengan Memodifikasi Algoritma Rivest Shamir Adleman (RSA). *Compiler, 3*(1), 61-74.

[9] Anita, F. (2018). Implementasi Algoritma Modular Multiplication Based Block Cipher dalam mengamankan Data Teks. *MEANS, 3*(2), 121-125.

[10] Wintolo, H., Retnowati, N. D., & Fendriyanto, P. (2013,). Penerapan Algoritma Lipat pada Steganografi yang Memanfaatkan RMS (Record Management System) di J2M2. *SENATIK STT Adisutjipto Yogyakarta, Vol.1*, pp. 76-84.