INVOTEK
Jurnal Inovasi Vokasional dan Teknologi

Volume 20 Number 1, 2020
ISSN: 1411 – 3411 (p) ISSN: 2549 – 9815 (e)
DOI: 10.24036/invotek.v20i1.647

Analisis Sistem Kriptografi dalam Mengamankan Data Pesan Dengan Metode One Time Pad Chiper

Oskah Dakhi[1*],Mardhiah Masril[2], Rina Novalinda[3], Jufrinaldi[4] dan Ambiyar[5]
[1]Teknik Informatika, Fakultas Teknik, STMIK Budidarma Medan
[2]Fakultas Ilmu Komputer, Universitas Putra Indonesia YPTK Padang
[3]Akademi Refraksi YLPTK Padang
[4]Seni Rupa Murni, Fakultas Seni Rupa dan Design, Institut Seni Indonesia Padang Panjang
[5]Teknik Mesin, Fakultas Teknik, Universitas Negeri Padang
[*]Corresponding author, e-mail: pemdakabnisel@gmail.com

Abstrak-- Dalam dunia teknologi informasi sering memanfaatkan komputer menjadi alat/media, sehingga keamanan pesan/data menjadi bagian utama pada sistem teknologi informasi. Sering kali informasi/pesan yang hendak kita kirimkan terancam keamanan/kerahasiaannya, karena diperlukan suatu metode yang bias membantu penyelesaian hal/masalah tersebut. Algoritma kriptografi One Time Pad (OTP) ialah suatu metode yang diaplikasikan dalam melindungi keamanan data/pesan. Pemilihan OTP dikarenakan algoritmanya sederhana/mudah serta belum bisa terselesaikan/ terpecahkan. Penggunaan kunci dalam algoritma ini wajib dibentuk secara acak agar data terjaga kerahasiaannya, maka perlunya penggunaan metode Linear Feeddback Shift Register (LFSR). Hasil dalam studi ini yaitu dihasilkannya sebuah aplikasi yang bisa dimanfaatkan dalam mengenkripsikan serta mendekripsi plaintext melalui algoritma OTP dan LCG/LFSR selaku pembangkit kunci.

Kata Kunci: : (OTP), Linear Feedback Shift Register (LFSR).

Abstract— ***In the world of information technology often utilizes computers to be tools/media, so message/data security becomes a major part of information technology systems. Often the information/messages that we want to send are threatened with security/confidentiality because we need a method that can help resolve these issues/problems. One Time Pad (OTP) cryptographic algorithm is a method that is applied in protecting data/message security. OTP is chosen because the algorithm is simple/easy and cannot be resolved/solved. the use of keys in this algorithm must be formed randomly so that data is kept confidential, it is necessary to use the Linear Feedback Shift Register (LFSR) method. The results of this study are an application that can be used to encrypt and decrypt plaintext through OTP and LCG / LFSR algorithms as key generators.***

Keywords: (OTP), Linear Feedback Shift Register (LFSR).

I. PENDAHULUAN

Pada lembaga atau institusi sangat memerlukan sebuah keamanan teknologi informasi yang baik untuk mengamankan aset terpentingnya seperti informasi dan mengamankan sistem keamanan komunikasi dari berbagai macam ancaman yang bisa muncul [1]. Sebagian teknik untuk melindungi komunikasi yaitu dengan mengimplementasikan teknik penyandian. Kriptografi merupakan pengetahuan dan seni untuk melindungi kerahasiaan pesan (data atau informasi) dengan teknik merahasiakan ke dalam bentuk kode yang tidak memiliki arti. [2] Dengan demikian, dibutuhkan sebuah sistem perlindungan data yang digunakan untuk mengamankan data. Sistem pengamanan ini menggunakan kriptografi one-time pad bisa dibuktikan sulit dipecahkan (lihat Claude Shannon "Teori komunikasi dari Sistem Kerahasiaan") (Claude Shannon 2012. Teknik pengkodean OTP mula-mula diperkenalkan oleh Gilbert Vernam pada perang dunia pertama. [3][4] Terdapat dua konsep penting dalam kriptografi yakni enkripsi dan dekripsi. Enkripsi ialah suatu metode pengamanan informasi/pesan yang akan

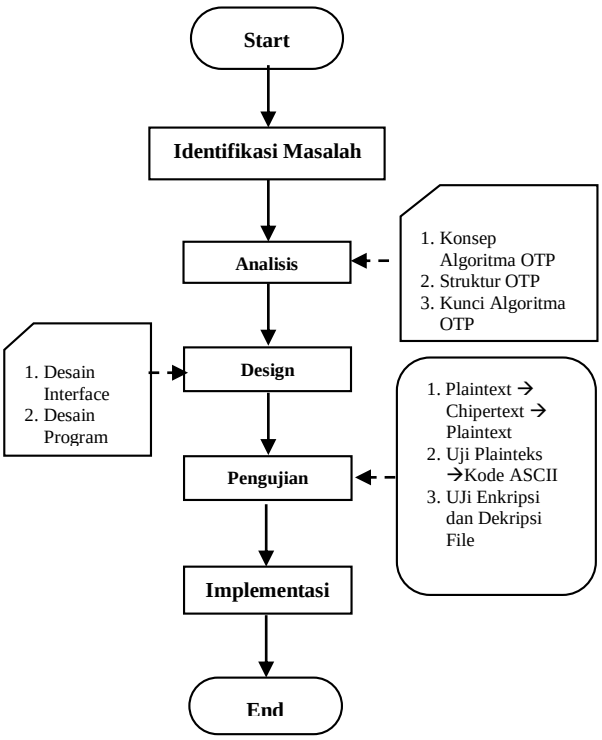
Received date 10/12/2019, Revised date 23/12/2019, Accepted date 23/01/2020

dikirimkan dengan mengubah format/bentuknya selaku informasi awal melalui penggunaan algoritma khusus agar tidak dapat terbaca oleh siapapun selain pengirim dan penerima informasi/pesan tersebut. Sedangkan dekripsi ialah kesebalikan dari enkripsi, yakni mengkonversi ulang irnformasi/pesan menjadi format/bentuk awalnya. [5][6][7].

Kriptografi ialah ilmu yang mempelajari bagaimana menjaga keamananan suatu pesan (plaintext). Tugas utama kriptografi adalah melakukan penjagaan pesan/kunci maupun keduanya agar tetap terjaga kerahasiaannya dari penyadap (attacker). Penyadap pesan diasumsikan mempunyai akses yang lengkap dalam saluran komunikasi antara pengirim pesan dan penerima pesan. [11]

II. METODE

Metode penelitian yang digunakan yakni *System Development Life Cycle* (SDLC) yang komprehensif dengan alat/media dan teknik yang digunakan pada sistem, sehingga hasil kajian yang dikembangkan memperoleh sistem yang terstruktur dapat didefinisikan dengan benar dan jelas serta membangun sistem *Flowchart* untuk mengilustrasikan sistem baru yang hendak dikembangkan menurut logika .[8]. Permasalahan yang akan diulas sebagaimana terlihat pada gambar 1.



Gambar 1. Kerangka Kerja

A. Analisa Sistem

Penelitian yang berjudul "Analisis Sistem Kriptografi Dalam Mengamankan Data Pesan Dengan Metode One Time Pad" yang akan mengenkripsinya dengan mengubah nilai pesannya menjadi karakter ASCII dan di XOR kan sehingga menghasilkan chiperteks pesan tersebut dengan aman.

B. Proses Enkripsi dan Dekripsi OTP

Terdapat dua ketentuan yang wajib dipenuhi dalam merancang unbreakable chiper yakni pemilihan kunci harus acak/random dan harus sama panjang dengan plainteks yang hendak dienkripsi. kedua ketentuan itu dapat berakibat pada plainteks yang serupa/sama belum pasti dienkripsikan menjadi chiperteks yang serupa atau sama.

Misalnya plainteks OSKAH dengan kunci DACHI. perlu dipahami bahwa plainteks panjang kunci harus sama demikian pula sebaliknya. pertama, kode ASCII serta plainteks harus didapatkan dahulu, kemudian dipindai ke biner sebagimana tergambar pada tabel 1.

Tabel 1. Notasi Biner Plainteks

Huruf	ASCII	Notasi Biner
O	78	0100 1111
S	83	0101 0011
K	75	0100 1011
A	65	0100 0001
H	72	0100 1000

Dari tabel 1 diatas menghasilkan kode ASCII Nota Biner Plainteks, begitu juga perlu dilaksanakan pada kunci yang dipilih.

Tabel 2. Notasi Biner Kunci

Huruf	ASCII	Notasi Biner
D	68	0100 0100
A	65	0100 0001
C	67	0100 0011
H	72	0100 1000
I	73	0100 1001

Dari table 2 masing-masing menghasilkan huruf yang telah di X0R-kan dengan kunci.

P: O = 0100 1111 S = 0101 0011 K = 0100 1011
A = 0100 0001 H = 0100 1000
K: D = 0100 0100 A = 0100 0001 C = 0100 0011
H = 0100 1000 I = 0100 1001
XoR==========
C: =0000 1011 0001 0010 0000 1000 0000 1001 0000 0001
ASCII : VT DC2 BS HT SOH

Tabel 3. Hasil X0R Plainteks dan Kunci

Chiperteks		
0000 1011	=	VT
0001 0010	=	DC2
0000 1000	=	BS
0000 1001	=	HT
0000 0001	=	SOH

Dari table 3 merupakan hasil proses dekripsi pesan X0R antara Cipher dengan kunci.

C: =0000 1011 0001 0010 0000 1000 0000 1001 0000 0001

K: =0100 0100 0100 0001 0100 0011 0100 1000 0100 1001

XOR

==========

P: =0100 1111 0101 0011 0100 1011 0100 0001 0100 1000

ASCII : O S K A H

Tabel 4. Hasil X0R Notasi Biner Dekripsi Data

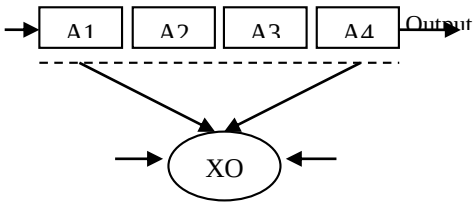
Plainteks	Kunci	Chiperteks
VT = 0000 1011	D= 0100 0100	O= 0100 1111
DC2=0001 0010	A= 0100 0001	S = 0101 0011
BS= 0000 1000	C= 0100 0011	K= 0100 1011
HT= 0000 1001	H = 0100 1000	A= 0100 0001
SOH=0000 0001	I = 0100 1001	H= 0100 1000

Dari table 4 menghasilkan nilai notasi biner yang telah di XOR-kan terhadap dekripsi data.

D. Generator Kunci

Untuk sekedar hendak menganalisis LFSR karena LFSR dipakai pada kriptografi dan teori pengandian telah difungsikan militer waktu dimulainya pemakaian alat elektronik prosedur algoritma U.[9][10]

Linear Feedback Shift Register (LFSR) Saya akan merancang sebuah LFSR 4 bit dengan keluaran pada bit ke 1.



Gambar 2. LFSR X0R

Contoh pada gambar 2 menggambarkan rancangan dasar dari LFSR yang artinya ialah "Register geser dengan umpan balik linier". Metodenya sebagai berikut:

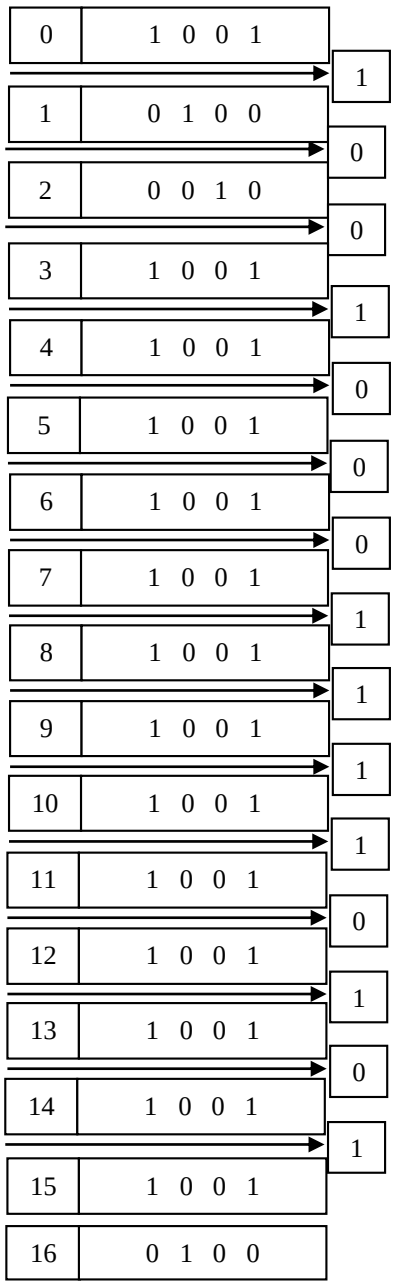
1). A1 sampai A4 diisikan dengan bit yang telah dipilih.
2). Tahapan pertama, A1 dan A4 akan diXoR-kan.
3). A1-A4 ditarik ke kanan sejauh satu bit.
4). Bit kesatu akan dijadikan *output.*
5). Bit perolehan XoR antara A1 dan A4 (sebelum ditarik) akan diinputkan ke A4.

Misalnya: diisikan A4 sampai A1 dengan angka 1001 dan buat register pergeseran sejumlah 16 kali. Berdasarkan pada gambar 2 *Linear Feedback Shift Register* (LFSR) XOR Output diatas hanya sejauh 4 bit, rentang waktu akan berulang ketika pergeseran/penarikan ke 15. Untuk menghitung rentang waktu maksimal sebuah LFSR digunakan formula seperti di bawah ini:

Periode = 2 ^ n - 1 n = Jumlah bit ^ = pangkat

Untuk merancang sebuah LFSR sejauh 8 bit maka rentang waktu maksimalnya yaitu 255.

Tahapan ke- A4 A3 A2 A1 ⟶ Output



Gambar 3. LFSR XOR Output

E. Algoritma Aplikasi

Adapun algoritmanya adalah sebagai berikut.

Langkah-langkah algoritmanya sebagai berikut.

1. Masukkkan plainteks kemudian tekan tombol acak kunci untuk mencari nilai kunci enkripsi dan dekripsi.
2. Randomize Timer dengan proses sebagai berikut.
   a. Mengosongkan tempat bit dengan proses untuk menghasilkan kunci acak dengan proses ini: ((255 – 32 + 1) * Rnd + 32).
   b. Kemudian setelah tempat bit kosong, dimasukkan bit-bit baru secara acak bit hasil proses acak tersebut merupakan kunci *(key)*.
3. Lakukan proses enkripsi dengan menekan tombol (Proses Enkripsi).
4. Hasil Biner digunakan sebagai kunci baru (Proses Enkripsi dan Dekripsi)
5. Periksa total hurup pada string, kunci, plainteks yang hendak dienkripsikan.
6. Lakukan pengulangan sejumlah total huruf, plainteks *dan string* tersebut
7. Modulo-kan huruf pertama beserta panjang kunci plainteks yang di *XoR*-kan dengan jumlah hasil terdahulu dan huruf setelahnya sampai yang terakhir hendak di *XoR* untuk mendapatkan chiperteks hasil enkripsi terdahulunya digunakan pada saat mengenkripsikan.

F. Tampilan Aplikasi Program

Rancangan tampilan aplikasi program ini digunakan untuk menjalankan program utama dari aplikasi ini yaitu men-*enkripsi* pesan yang diinginkan dengan menggunakan algoritma *Vernam*/OTP. Sehingga menghasilkan *chipertext* (pesan yang telah di-*enkripsi*.

Gambar 4. Rancangn Aplikasi Program

III. HASIL DAN PEMBAHASAN

Implementasi sistem ini menjelaskan mengenai skenario *test*, yang digunakan dalam langkah-langkah implementasi perangkat lunak sistem OTP. Pengujian sistem menjelaskan tentang pengujian hasil pengenkripsi dan pengdekripsian data pesan berupa *file* dengan metode *one time pad*. Seperti dijelaskan dalam penelitian terdahulu bahwa penggunaan perangkat ini untuk menjaga kerahasiaan pesan yang ingin dikirim [11]12]

3.1. Skenario Test

Langkah awal untuk mensimulasikan skenario ini adalah menjalankan aplikasi OTP, pilih tombol untuk enkripsi kemudian *browse* lokasi dimana *file* yang akan di enkripsi dan masukkan *password* serta konfirmasi *password* sehingga *password* yang dimasukkan nantinya digunakan untuk mendekripsikan *file* yang telah dienkripsi, begitu juga sebaliknya untuk mendapatkan plainteksnya atau *file* aslinya harus di masukkan kunci dekripsinya sehingga menghasilkan pesan (*file*) atau plainteks awal.

$$C = (P \oplus K) \quad (1)$$

Berikut merupakan logika matematika algoritma *one-time pad* dalam proses mengenkripsi, mengacak sebuah kunci serta mengdekripsikan sebuah *file.*

Contoh: dimasukkan pesan dan *key* maka chiperteksnya seperti di bawah ini.

Plainteks : Oskar Dakhi 122321060
Kunci : UÖó2µíéO_kG²ôÔ□AK¤?SŽ
Chiperteks: ¥˜SÇÍ-.4[].'Åæ½ry•e¾

Pada proses "Enkripsi Teks" di atas dijalankan melalui proses sebagai berikut:

1). Pertama Plainteks diubah ke dalam bentuk kode ASCII, kemudian diubah ke dalam notasi biner sebagai berikut:

Plainteks: Oskar Dakhi 122321060

ASCII Desimal Pesan:

(79,115,107,97,114,32,68,97,107,104,105,32,49,50,50,51,50,49,48,54,48)

O (79), **s** (115), **k** (107), **a** (97), **r** (114), (**spasi**) 32, **D** (68), **a** (97), **k** (107), **h** (104), **i** (105), (**spasi**) 32, **1** (49), **2** (50), **2** (50), **3** (51), **2** (50), **1** (49), **0** (48), **6** (54), **0** (48).

Tabel 5 Notasi Biner Plainteks

Karakter	ASCII	Notasi Biner
O	0079	0100 1111
S	0115	0111 0011
K	0107	0110 1011
A	0097	0110 0001
R	0114	0111 0010
Spasi	0032	0010 0000
D	0068	0100 0100
A	0097	0110 0001
K	0107	0110 1011
H	0104	0110 1000

I	0105	0110 1001
Spasi	0032	0010 0000
1	0049	0011 0001
2	0050	0011 0010
2	0050	0011 0010
3	0051	0011 0011
2	0050	0011 0010
1	0049	0011 0001
0	0048	0011 1000
6	0054	0011 0110
0	0048	0011 1000

Pada tabel 5 diatas menghasilkan nilai ASCII melalui enkripsi pesan.

2). Setelah itu kunci juga diubah dalam bentuk kode ASCII, kemudian diubah ke dalam notasi biner sebagai berikut:
Kunci: UÖó2µíéO_kG²ôÔ□AK¤? SŽ
ASCII Desimal Pesan:
(85,214,243,50,181,237,233,79,95,107,71,178,244,212,143,65,75,164,63, 83,142)
Notasi Biner: lihat tabel 6.

Tabel 6. Notasi Biner Kunci

Karakter	ASCII	Notasi Biner
U	0085	0101 0101
Ö	0214	1101 0110
Ó	0243	1111 0011
2	0050	00110010
µ	0181	1011 0101
Í	0237	1110 1101
É	0233	1110 1001
O	0079	0100 1111
_	0095	0101 1111
K	0107	0110 1011
G	0071	0100 0111
²	0178	1011 0010
Ô	0244	1111 0100
Ô	0212	1101 0100
	0143	1000 1111
A	0065	0100 0001
K	0075	0100 1011
¤	0164	1010 0100
?	0063	0011 1111
S	0083	0101 0011
Ž	0142	1000 1110

Pada tabel 6 diatas menghasilkan nilai notasi biner melalui pengubahan kode ASCII dengan enkripsi pesan.

Kunci yang dibuat oleh program tersebut harus acak agar tidak bisa dipecahkan karena kerahasiaan pesan tetap terjaga, kunci tersebut menggunakan pembangkit aliran yang menyesuaikan dengan panjang dari plainteks. Hal ini sesuai dengan yang teah diterangkan dalam penelitian terdahulu [13] [14].

3). Kemudian masing-masing huruf plainteks diatas diXoR-kan beserta kunci, kemudian perolehan dari Enkripsi tersebut adalah sebagai berikut:
Chiperteks: ¥˜SÇÍ-.4[].’Åæ½ry•e¾
ASCII Desimal Pesan:
(26,165,152,83,199,205,173,46,52,3,46,146,197,230,189,114,121,149,15, 101,190,)

Tabel 7 Perolehan XoR Plainteks dan Kunci

Karakter	ASCII	Notasi Biner
	0026	0001 1010
¥	0165	1010 0101
˜	0152	1001 1000
S	0083	0101 0011
Ç	0199	1100 0111
Í	0205	1100 1101
-	0173	1010 1101
.	0046	0010 1110
4	0052	0011 0100
	0003	0000 0011
.	0046	0010 1110
’	0146	1001 0010
Å	0197	1100 0101
Æ	0230	1110 0110
½	0189	1011 1101
R	0114	0111 0010
Y	0121	0111 1001
•	0149	1001 0101
	0015	0000 1111
E	0101	0110 0101
¾	0190	1011 1110

Dari tabel 7 diatas menghasilkan huruf plainteks diatas diXoR-kan beserta kunci, yang perolehan dari Enkripsi pesan.

4. Berikut hasil dekripsinya seperti pada tabel 8. berikut.

$$P = (C \oplus K) \quad (2)$$

Tabel 8. Notasi Biner Dekripsi Data

Plainteks	Kunci	Notasi Biner
= 0001 1010	U = 0101 0101	O = 0100 1111
¥ = 1010 0101	Ö = 1101 0110	s = 0111 0011
˜ = 1001 1000	ó = 1111 0011	k = 0110 1011
S = 0101 0011	2 = 00110010	a = 0110 0001
Ç = 1100 0111	µ = 1011 0101	r = 0111 0010
Í = 1100 1101	í = 1110 1101	Spasi =0010 0000
- = 1010 1101	é = 1110 1001	D = 0100 0100
. = 0010 1110	O = 0100 1111	a = 0110 0001
4 = 0011 0100	_ = 0101 1111	k = 0110 1011
ETX= 0000 0011	k = 0110 1011	h = 0110 1000
. = 0010 1110	G = 0100 0111	i = 0110 1001
’ = 1001 0010	² = 1011 0010	Spasi = 0010 0000
Å = 1100 0101	ô = 1111 0100	1 = 0011 0001
Æ = 1110 0110	Ô = 1101 0100	2 = 0011 0010
½ = 1011 1101	1000 1111	2 = 0011 0010
r = 0111 0010	A = 0100 0001	3 = 0011 0011
y = 0111 1001	K = 0100 1011	2 = 0011 0010
• = 1001 0101	¤ = 1010 0100	1 = 0011 0001
SI= 0000 1111	? = 0011 1111	0 = 0011 1000
E = 0110 0101	S = 0101 0011	6 = 0011 0110
¾ = 1011 1110	Ž = 1000 1110	0 = 0011 1000

Dari tabel 8 diatas menghasilkan huruf plainteks diatas diXoR-kan beserta kunci, yang perolehan dari Deskripsi pesan

3.2 Implementasi Sistem

Proses implementasi program merupakan penerapan hasil dari perancangan yang telah dibuat dengan membangun program aplikasi sistem *one time pad* (OTP) dalam penyandian pesan (*file*) dan berikut contoh kasusnya pada (*file)* :

Gambar 5. Tampilan Program Penyandian

Adapun proses enkripsinya adalah sebagai berikut:

a. Proses Enkripsi Pesan (*File*)

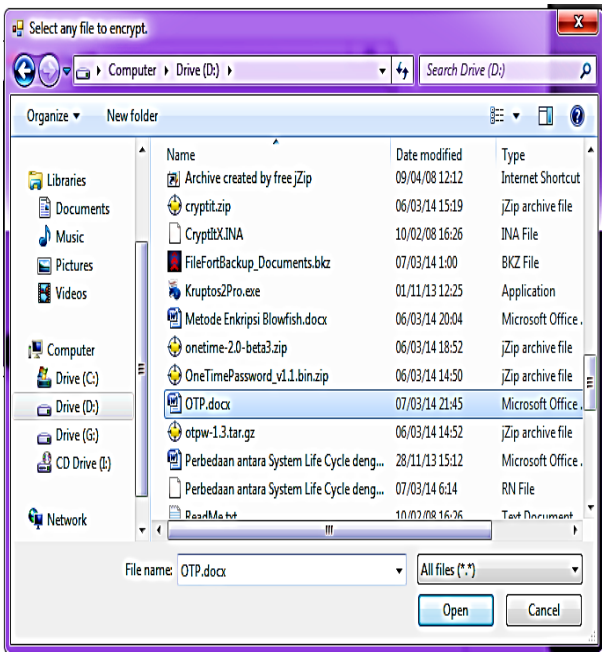
Gambar 6. Proses Enkripsi Pesan (*File*)

Dari gambar 6 bisa dilihat bahwa proses penyandian pesan dengan tahapan sebagai berikut:

1. Pilih tombol "Encrypt File" maka tampilan gambar 7 *Start Encryption.*

Gambar 7. Tampilan Start Encryption

2. Pilih "*Bwowse"* untuk memilih *file* yang akan di enkripsi maka tampil gambar 8 berikut.

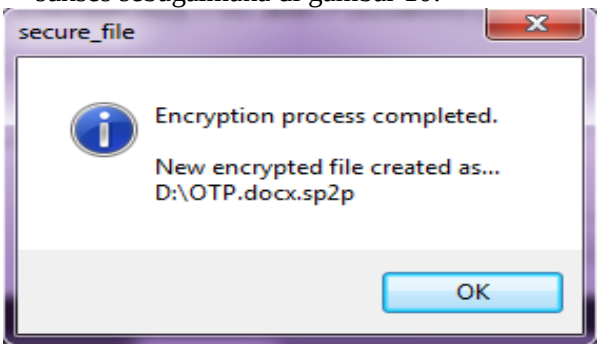


Gambar 8. *Path* Pemilihan *File* Enkripsi

3. Kemudian masukkan *Password* dan *Confirm Password* yang digunakan nantinya untuk mengdekripsi *file* yang telah di enkripsi.

Gambar 9. Proses Memasukkan Password dan Konfimasi Password

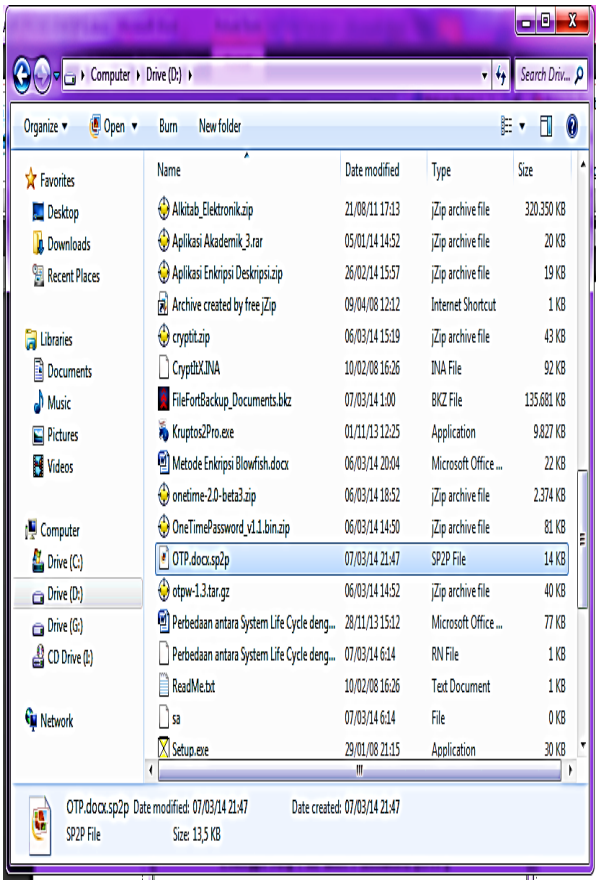
4. Klik "Start Encryption" untuk melakukan proses enkripsi pada *file*.
5. Setelah di enkripsi kemudian muncul hasil sukses sebagaimana di gambar 10.



Gambar 10. Proses Enkripsi Sukses

6. Klik tombol "OK" untuk kembali ke aplikasi.

7. Berikut tampilan *file* yang telah terenkripsi oleh aplikasi OTP sebagaimana pada gambar 11 di bawah.

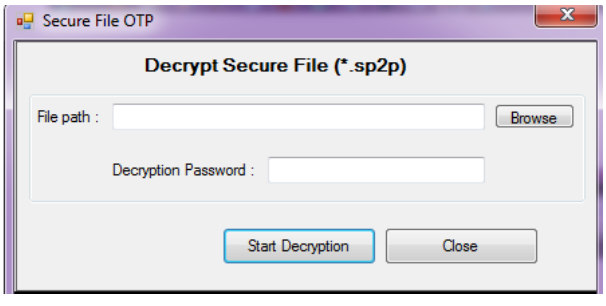


Gambar 11. Hasil Enkripsi

b. Proses Dekripsi Pesan (*File*)

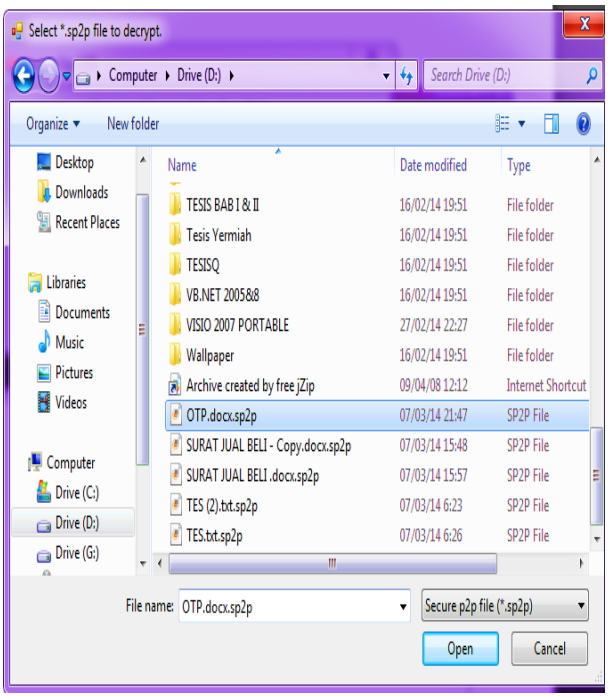
Pada enkripsi hanya saja tidak lagi ada memasukkan konfirmasi *password* tetapi hanya memasukkan *password* yang telah ditentukan oleh pengenkripsi *file* pada kotak *password* [15]. Berikut langkah-langkahnya:

1. Pilih tombol "Decrypt File" maka tampilan gambar 12 *Start Decryption.*



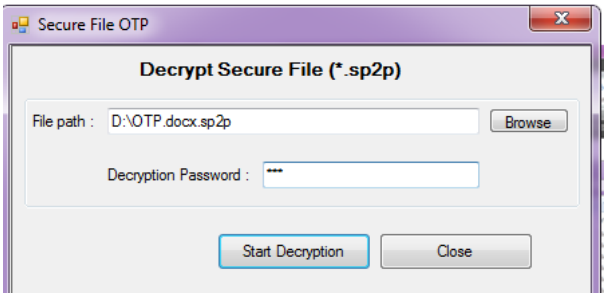
Gambar 12. Tampilan Start Decryption

1. Pilih "*Bwowse"* untuk memilih *file* yang akan di dekripsi maka tampil gambar 13.



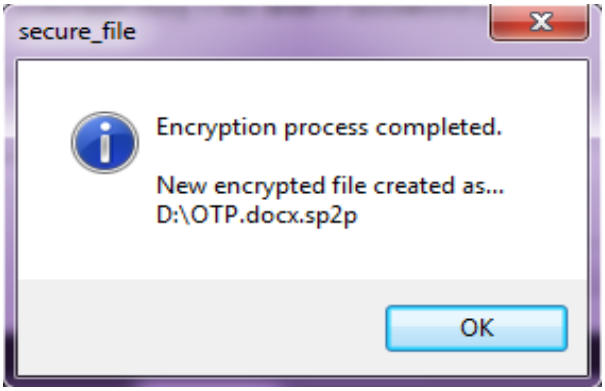
Gambar 13. *Path* Pemilihan *File* Dekripsi

2. Kemudian masukkan *Decryption Password* pada kotak *password* untuk membuka *file* yang telah dienkripsi seperti gambar 14:



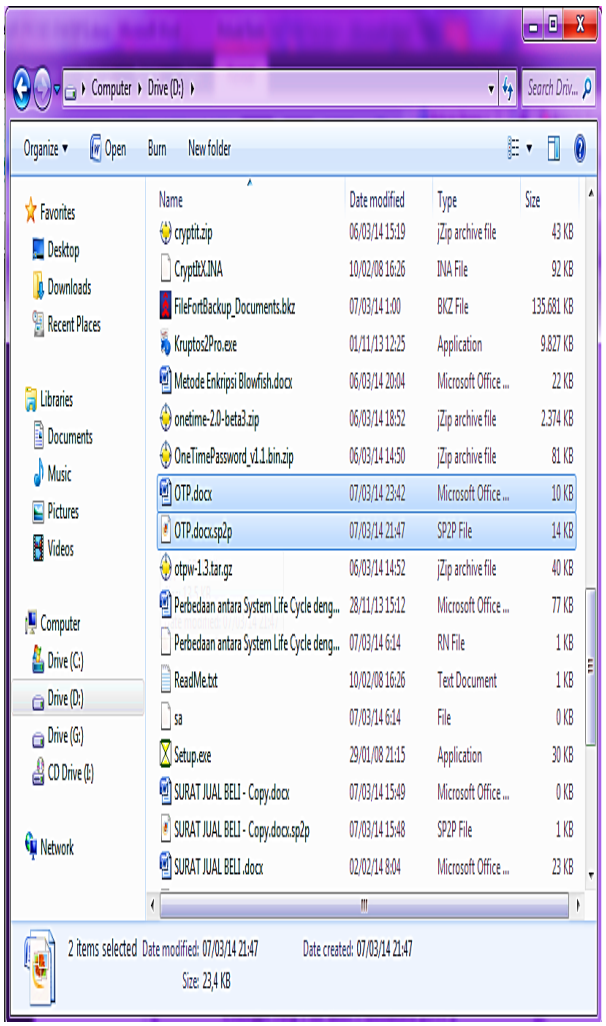
Gambar 14. Proses Memasukkan Password/Kunci

3. Klik "Start Decryption" untuk melakukan proses dekripsi pada *file*.
4. Setelah di dekripsi maka akan tampil hasil sukses sebagaimana di gambar 15.



Gambar 15. Dekripsi Sukses

6. Klik tombol "OK" untuk keluar dari aplikasi.
7. Berikut hasil dari proses dekripsi pada *file* seperti pada gambar 16.



Gambar 16. Hasil Dekripsi pada *File*

IV. KESIMPULAN

Setelah melakukan implementasi dan pengujian kemudian yang merupakan kesimpulan dalam penelitian ini sebagai berikut: Proses teknik One Time Pad Chiper dalam mengenkripsi dan dekripsikan data pesan (file) sehingga sulit di terjemahkan orang yang tidak berhak yaitu dengan mengubah nilai kebentuk karakter ASCII kemudian diubah kebentuk biner setelah itu di XOR-kan dengan plainteks dengan kunci maka akan menghasilkan sebuah chiperteks yang terenkripsi. Metode One Time Pad Chiper dalam mengamankan data pesan (file) yaitu dengan mengenkripsi file dengan mengubah ekstensi dan nilai karakternya sehingga sulit untuk diterjemahkan oleh kriptanalisis.

DAFTAR PUSTAKA

[1] Ariyus, Dony. Kriptografi Keamanan Data dan Komunikasi.Yogyakarta: Graha Ilmu, 2006.

[2] Rinaldi Munir (2006). "Kriptografi." Bandung: Penerbit Informatika. 1-199.

[3] Shannon, Claude. "Teori Komunikasi Sistem Kerahasiaan", *Jurnal Teknis Sistem Bell* , vol. 28 (**4**), halaman 656-715, 1949.

[4] Bilqis (2012). "Analisis dan Perancangan Aplikasi Pesan Rahasia menggunakan Algoritma One Time Pad (OTP) dengan Pembangkit Bilangan Acak Linear Congruential Generator (LCC)." Universitas Sumatera Utara : Skripsi.

[5] Mulyono, Hengky. 2013. "Implementasi Algoritma *One Time Pad* pada Penyimpanan Data Berbasis *Web"*. Gunadarma,.

[6] Rifki Sadikin (2012). "Kriptografi Untuk Keamanan Jaringan." Yogyakarta: Penerbit Andi. 15-54.

[7] Dony Ariyus (2008). "Pengantar Ilmu Kriptografi Teori, Analisis, dan Implementasi." Yogyakarta: Andi Yogyakarta. 43-45.

[8] Andrew Fiade, (2010), Jurnal : Usulan Perkembangan Metodologi SDLC Untuk Sistem Informasi Web, Teknik Informatika, Fakultas Ilmu Komputer, Universitas Mercu Buana, Jakarta.

[9] Rinaldi Munir. "Message Authentication Code(MAC)." http://www.informatika.org/~rinaldi/Kriptografi/MAC%20dan%20Random%20Generator.pdf. Diakses tanggal 17 Februari, 2014.

[10] Yudi Haribowo. "Kriptanalisis Terhadap Pembangkit Bilangan Acak Semu."

[11] Achmad Fauzi, Yani Maulita, Novriyenni. (2016). "*Analisis Super Enkripsi Algoritma One Time Pad Dan Algoritmaelgamal Pada Keamanan Pesan"* Prosiding Seminar Nasional Inovasi dan Teknogi Informasi. Hal 350-357.

[12] Muhammad Khoiruddin Harahap, Nurul Khairina (2017). *"Analisis Algoritma One Time Pad Dengan Algoritma Cipher Transposisi Sebagai Pengamanan Pesan Teks"* Jurnal & Penelitian Teknik Informatika; 1(2); 58-62.

[13] Endah Pratiwi Lis. 2014. "*Program Aplikasi Kriptografi Penyandian One Time Pad Menggunakan Sandi Vinegere"*, Universitas Pendidikan Indonesia.

[14] Tomoyud S Waruwu, Kristina Telaumbanua. 2016. *"Kombinasi Algoritma OTP dan Algoritma BBS dalam Pengamanan File"*, Mikroskill. Medan.

[15] Erna Kumalasari Nurnawati. 2008. *"Analisis Kriptografi menggunakan Algoritma Vigenere Cipher dengan Mode Operasi Cipher Block Chaining (CBC)"*, IST Akprind.

Biodata Penulis

Oskah Dakhi, Lahir di Nias, 17 Februari 1989. Sarjana Komputer di Jurusan Teknik Informatika STMIK Budidarma Medan 2012. Tahun 2014 memperoleh gelar Magister Ilmu Komputer di jurusan Teknik Informatika Pascasarjana UPI YPTK dengan bidang konsentrasi Teknik Informatika. Mahasiswa Program Doktor di jurusan pendidikan dan teknologi kejuruan FT UNP sejak tahun 2019- sekarang.

Mardhiah Masril, Lahir di Bengkulu, 12 Oktober 1984. Magister Ilmu Komputer di jurusan Teknologi Informasi Pascasarjana UPI YPTK dengan bidang konsentrasi Teknologi Informasi. Mahasiswa Program Doktor di jurusan pendidikan dan teknologi kejuruan FT UNP sejak tahun 2019-sekarang.

Rina Novalinda, Lahir 25 April 1972. Sarjana Teknik dan Manajemen Industri Universitas Islam Bandung, Tamat 1996. Tahun 2004 memperoleh gelar Magister Manajemen, Kosentrasi MSDM Universitas Budi Luhur Jakarta. Mahasiswa Program Doktor di Jurusan Pendidikan Teknologi dan Kejuruan FT UNP 2019-Sekarang.

Jufrinaldi, Lahir di Bukit tinggi 26 Desember 1969. Memperoleh gelar sarjana S1(S. Sn) Program Studi Seni Rupa Murni di ISI Yogyakarta tahun 1996. Memperoleh Gelar Magister Seni (M. Sn) di Program Studi Pengkajian Seni Rupa Pasca Sarjana ISI Padang panjang tahun 2015. Sedang menempuh Program Doktor di Jurusan Pendidikan Teknologi dan Kejuruan sejak tahun 2019 sampai sekarang. Pekerjaan Dosen di FSRD ISI Padang panjang dari tahun 1997 sampai sekarang.

Ambiyar, Lahir 13 Februari 1955 di Padang Padang Sumatera Barat. Sarjana Teknik Mesin di Fakultas Keguruan Teknik (FKT) IKIP Padang 1979. Tahun 1986 memperoleh gelar Magister di IKIP Yogyakarta dan Jakarta, Tahun 2005 lulus Program Doktor di Universitas Negeri Jakarta (UNJ), Diangkat sebagai dosen di jurusan Pendidikan Teknik Mesin FT UNP sejak tahun 1981- sekarang.