Implementasi Algoritma LUC Dalam Penyandian Teks

Dewi Ramadani
STMIK Budi Darma Medan, Jl. SM.Raja No.338 Sp.Limun Medan, Sumut, Indonesia.
Email : dewiramadani11@gmail.com

ABSTRACT
Confidentiality and data security are very important in data communication, both for the purpose of shared security and for individual privacy. Computer users who want the data to be unknown by unauthorized parties always try to get around the way to secure the information that will be communicated or to be stored. One method for securing data or information is cryptographic methods. The LUC algorithm is a cryptographic method using two different keys in the cryptosystem. To encrypt text, an encryption function is used that uses a public key, the result of encryption is encrypted text that is safe from intruders. Furthermore, by decrypting encrypted text using the decryption function using the private key it will return the same text as the original. With the LUC algorithm the author tries to create a text encoding to secure data with the LUC algorithm.

Kata Kunci: Kriptografi, Penyandian Teks, Algoritma LUC.

PENDAHULUAN

Kriptografi tidak hanya menyediakan alat untuk keamanan pesan, tetapi juga sekumpulan teknik yang berguna[7]. Kriptografi didefinisikan sebagai ilmu sekaligus seni untuk menjaga kerahasiaan pesan (data atau informasi) dengan cara menyamarkannya menjadi bentuk tersandi yang tidak mempunyai makna bagi yang tidak memahaminya [4].

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 menyiasati cara mengamankan informasi yang akan dikomunikasikan atau yang akan disimpan [11].

Banyak jenis tentang algoritma kriptografi untuk mengamankan pesan rahasia. Salah satu algoritma yang digunakan dalam mengamankan data pesan yaitu menggunakan algoritma LUC, algoritma tersebut menggunakan dua buah kunci yaitu kunci umum (untuk melakukan enkripsi) dan kunci rahasia (untuk melakukan dekripsi). Operasi pada LUC dilakukan dalam domain bilangan, oleh karena itu sebelum dilakukan *enkripsi*¸ teks terlebih dahulu dikonversi ke dalam bentuk angka[10]. Algoritma LUC sebenarnya hampir sama dengan metode kriptografi yang lain yaitu metode RSA (Rivest, Shamir, Adleman), hanya saja fungsi pangkat pada metode RSA diganti fungsi Lucas dimana pertambahan nilai barisan Lucas sampai dengan *n* suku sangat cepat, sehingga dikembangkan fungsi modulo N>2.

LANDASAN TEORI

2.1 Implementasi

Secara etimologis pengertian implementasi menurut Webster adalah Implementasi yang merupakan terjemahan dari kata *implementation*, berasal dari kata kerja *to implement*. Kata *to implement* berasal dari bahasa Latin *implementum* dari asal kata *impere* dan *plere*. Kata *implere* dimaksudkan *to fil up, to fil in,* yang artinya mengisi penuh, melengkapi sedangkan *plere* maksudnya *to fil* yaitu mengisi.[9].

Selanjutnya kata *to implement* dimaksudkan sebagai :

1. *To implement* dimaksudkan membawa ke suatu hasil (akibat), melengkapi dan menyelesaikan.
2. *To implement* dimaksudkan menyediakan sarana (alat) untuk melaksanakan sesuatu, memberikan hasil yang bersifat praktis terhadap sesuatu.
3. *To implement* dimaksudkan menyediakan atau melengkapi dengan alat [8].

Jadi secara etimologis implementasi itu dapat dimaksudkan sebagai suatu aktivitas yang bertalian dengan penyelesaian suatu pekerjaan dengan penggunaan sarana (alat) untuk melaksanakan sesuatu dan memperoleh hasil.

2.2 Kriptografi

Kriptografi (*Cryptography*) berasal dari bahasa yunani : "*cryptos*"artinya "*secret*" (rahasia) sedangkan "*graphein*" artinya "*writing*" (tulisan). Jadi kriptografi berati

"*secret writing*"(tulisan rahasia). Secara umum kriptografi merupakan ilmu dan seni untuk menjaga kerahasian dari suatu pesan. Kriptografi juga dapat diartikan sebagai ilmu yang mempelajari teknik-teknik matematika yang berhubungan dengan aspek keamanan informasi seperti kerahasian, integritas data, serta otentikasi [7].

2.3 Algoritma LUC

Algoritma Luc merupakan metode kriptografi dengan menggunakan dua kunci yang berbeda dalam kriptosistemnya [10]. Untuk mengenkripsi teks digunakan fungsi enkripsi yang menggunakan sebuah kunci publik, hasil enkripsi merupakan teks terenkripsi yang aman dari pihak yang tidak berhak atas informasi didalamnya.

Barisan Rantai Lucas merupakan sebuah bilangan bulat n, maka barisan Rantai Lucas dari n adalah (b0, b1, … , br) dimana b0 = (0 atau 1), b1 = (0 atau 1) dan br = 0. Untuk b0 hingga br, misalkan z = n mod 2, jika z = 1 maka (b0 = 1 dan n/2), selain itu jika z = 0 maka (b0 = 0 dan n/2). Pembagian tersebut dilakukan secara berulang hingga br dan pada akhirnya br = 0.

Algoritma Luc juga merupakan salah satu algoritma dalam kriptografi umum, algoritma dibangun berdasarkan fungsi matematika yang barisan Lucas yang telah didefenisikan sebelumnya : $f_{luc}(P) = V_n(P,1) \bmod N$.

Dalam menyelesaikan algoritma Luc terdapat tiga tahap utama yaitu algoritma pembangkitan kunci, proses enkripsi dan proses dekripsi.

1. Pembangkitan Kunci
   Dalam algortima LUC pada saat membangkitkan sepasang kunci membutuhkan dua buah bilangan prima p dan q.
   (1) N = p x q
   Dihitung nilai fungsi perluasan eurlerΦ (N)
   (2) Φ(N) = (p-1)(p+1)(q-1)(q+1)
   Sebuah bilangan bulat, e Є Z, 1 < e < Φ(N), yang disebut kunci enkripsi, kemudian dicari sedemikian sehingga e dan Φ(N) berelatif prima
   (3) Nilai (e,n) dipublikasikan sebagai kunci publik algoritma LUC. Setelah kunci publik diperoleh, langkah selanjutnya menghitung kunci dekripsi (kunci privat) diperoleh dengan terlebih dahulu menghitung nilai D (diskriminan) barisan LUC.
   (4) $D=m^2-4$, dimana m adalah plainteks yang akan dienkripsikan.
   (5) S(N) = KPK[ ]
   S(N) mempunyai empat kemungkinan yaitu:

$$\left(p-\frac{D}{p}\right).\left(q-\frac{D}{p}\right)$$

   S(N) = KPK [(p-1),(q-1)]
   S(N) = KPK [(p-1),(q+1)]
   S(N) = KPK [(p+1),(q-1)]
   S(N) = KPK [(p+1),(q+1)]
   Sehingga nilai kunci dekripsi d mempunyai empat kemingkinan tergantung dari nilai S(N), dan diperoleh dengan mencari invers perkalian modul S(N).
   (6) ed = 1 mod S(N)
   Nilai *d* diperoleh dengan cara berikut :
   e.d = 1 mod S (N)

$$d = \frac{1 + k.S(N)}{e}$$

   Dengan *k* adalah bilangan peubah sebarang nilai *(d,N)* yang diperoleh merupakan kunci dekripsi (kunci privat) dari kunci enkripsi *(e,N)*.
2. Proses Enkripsi
   Plainteks m akan dienkripsikan dengan kunci publik e yang diperoleh dari hasil pembangkit kunci. Fungsi enkripsi didefinisikan sebagai berikut :
   $f_{enk} = Ve\ (M,1) \bmod N$
   Fungsi enkripsi akan menghitung suku ke-n dari barisan rantai lucas yang disimpan dalam k[x] yang mempunyai nilai 0 atau 1 dengan indeks n adalah kunci publik e dan M adalah Plainteks, sehingga untuk mengenkripsi m Є M dan kunci publik (e,N) dinyatakan sebagai : $c = V_e\ (mi, 1) \bmod N$
3. Proses Dekripsi
   Cipherteks c Є C diperoleh dari proses enkripsi. Proses dekripsi kurang lebih sama dengan proses enkripsi, perintah yang digunakan pada tahap dekripsi ini sama dengan proses enkripsi hanya nilai *e* dalam proses enkripsi diganti dengan nilai *d* yang didapatkan pada proses pembangkitan kunci privat. fungsi dekripsi didefinisikan sebagai: $f_{enk}\ (C) = V_n\ (C,1) \bmod N$ untuk mendekripsikan c Є C dan kunci privat (d,N) untuk mendapatkan plainteks m Є M dinyata sebagai : $m = V_d(c,1) \bmod N$. [12].

2.4. Penyandian Teks

Penyandian teks adalah ilmu yang bersandarkan pada teknik matematika untuk berurusan dengan keamanan informasi seperti kerahasiaan, keutuhan data dan otentik entitas [6].

Teks adalah sederetan huruf, angka dan simbol-simbol khusus (misalnya + dan $) yang kombinasinya tidak tergantung pada masing-masing item secara individual, contoh teks adalah artikel Koran [1]. Teks juga didefinisikan sebagai bentuk media yang paling umum digunakan dalam menyajikan informasi, baik yang menggunakan model baris perintah ataupun *Graphic User Interface* (GUI), teks juga dapat disajikan dengan berbagai bentuk *font* maupun [2].

HASIL DAN PEMBAHASAN

3.1 Analisa Algoritma LUC Dalam Menyandikan Teks

Dalam menyelesaikan algoritma LUC terdapat 3 tahap yaitu pembangkitan kunci, proses enkripsi dan proses dekripsi.

1. Pembangkitan Kunci
   a. Pembangkitan kunci public cukup sederhana karena untuk kunci *N* hanya dengan mengalikan dua bilangan prima yang telah ditentukan. Selanjutnya untuk menentukan nilai *e* maka dicari bilangan yang berelatif prima dengan (*p*-1), (*p*+1), (*q*-1), (*q*+1). Dengan menggunakan fungsi pembagian modulo maka pencarian bilangan yang relatif prima dapat ditemukan.
   b. Pembangkitan kunci privat adalah proses mencari nilai *d* dan membutuhkan kunci public *e*.
2. Proses enkripsi, tahap pertama dalam proses enkripsi ini adalah menentukan Barisan Rantai Lucas dan disimpan dalam *k[x]* yang mempunyai nilai 0 atau 1, tahap selanjutnya adalah proses dekripsi dengan menggunakan persamaan, tergantung nilai *k*[*x*].
3. Proses dekripsi kurang lebih sama dengan proses enkripsi, perintah yang digunakan pada tahap dekripsi ini sama dengan proses enkripsi hanya nilai *e* dalam proses enkripsi diganti dengan nilai *d* yang didapatkan pada proses pembangkitan kunci privat.Algoritma LUC pada dasarnya dibedakan menjadi tiga bagian besar, yakni algoritma pembangkit kunci, proses *enkrips* dan proses *dekripsi.*

3.1.1. Pembangkit Kunci Algoritma LUC

Langkah-langkah yang perlu dilakukan dalam proses pembentukan kunci, baik kunci publik atau kunci privat adalah sebagai berikut :

1. Pilih dua (2) buah bilangan prima p dan q.
   Misalkan nilai p adalah 47 dan q adalah 241, p dan q adalah bilangan prima, akan tetapi nilai p ≠ q.
2. Hitung n = p x q, sehingga nilai n = 47 x 241 adalah 11327
3. Hitung Φ(n) = (p-1) (p+1) (q-1) (q+1), sehingga nilai Φ(n) =
   RP (p-1) = RP (46) = {3,5,7,11,13,17,19,29,31,37,41,43}
   RP (p+1) = RP (48) = {5,7,11,13,17,19,23,29,31,37,41,43,47}
   RP (q-1) = RP (240) = {7,11,13,17,19,23,29,31,37,41,43,…… 239}
   RP (q+1) = RP (242) = {3,5,7,13,17,19,23,29,31,37, 41, 43,…….,241}
4. Pilih kunci publik (e), sehingga nilai e relatif prima tehadap Φ(n) adalah 7. Maka nilai e adalah 7 jadi nilai e,n (7,11327).
5. Hitung kunci privat (d) dengan menggunakan $D=m^2-4$ dimana m adalah plainteks yang akan dienkripsikan.
   $d=(6869)^2-4$ simbol legender untuk $\frac{D}{P}$ adalalah $\frac{57017597}{47}$ =-1, sedangkan simbol legendre untuk $\frac{D}{q}$ adalalah $\frac{57017597}{241}$ =1.
   Langkah berikutnya adalah mencari LCM $\left(p-\frac{D}{p}, q-\frac{D}{q}\right)$.
   S(n) =LCM(p+1,q-1)
   S(n) =LCM(47+1,241-1) = 240
   Perhatikan bahwa d e = 1 (mod n) sehingga e.d = $\frac{1+\mathbf{k.s(n)}}{e}$ dengan rumus tersebut maka di dapat nilai d = $\frac{1+3\mathbf{x240}}{7}$
   d = 103
   maka didapatkan kunci privat adalah (103,11327).

Pada kunci publik dijelaskan bahwa n, modulus yang digunakan dan e adalah kunci publik atau kunci untuk proses *enkripsi*. Sedangkan pada kunci privat dijelaskan bahwa n, modulus yang digunakan dan d, kunci privat, adalah kunci untuk *dekripsi*, yang harus dijaga kerahasiaannya.

3.1.2. Proses Enkripsi

Tahap awal pada proses enkripsi adalah mengatur teks menjadi blok-blok yang terdiri

dari dua karakter. Misal teks yang akan disandikan adalah "D E W I", apabila dipisahkan dalam blok maka teks berubah menjadi "DE" dan "WI". Selanjutnya adalah merubah tiap blok dalam bentuk ASCII, maka didapatkan bilangan ASCII DE = 6869 dan WI = 8773, kemudian dihitung dengan menggunakan fungsi lucas $C_i = V_e(m_i,1) \bmod N$. Dengan *e* = 7, *m*i adalah nilai ASCII tiap blok, *N* = 11327 dan *c*i adalah hasil enkripsi tiap blok. Dengan menggunakan kunci public yang dibangkitkan pada tahap sebelumnya (*e*, *N*) = (7, 11327), tentukan barisan Rantai Lucas dalam *k*[*x*].
perhitungan barisan rantai lucas adalah
e = 7 (dimana 7 adalah bilangan yang tidak habis dibagi 2), maka
e = 7 – 1 = 6 (6 adalah bilangan yang habis dibagi 2), maka
e = 6/2 = 3 dan seterusnya.

Tabel 1 Barisan Rantai Lucas *Enkripsi*

x	k[x]	e
	1	e-1 = 6
2	0	e/2 = 3
3	1	e-1 = 2
4	0	e/2 = 1

Didapatkan k[x] = {1,0,1,0} dimana k[x] adalah barisan rantai lucas maka k[x] = {0,1,0,1}. Proses enkripsi menggunakan rumus sesuai dengan nilai k[x]. untuk mendapatkan hasil perhitungan enkripsi digunakan rumus :
c_i = Ve (m, 1) mod N,
Adapun proses perhitungan enkripsi adalah sebagai berikut :
Plainteks (M): DEWI
DE : 6869
WI : 8773
c_i = Ve (m, 1) mod N
c_i = 2 (15662,1) mod 11327
= 31324 mod 11327
= 8670
c_i = 3 (15662,1) mod 11327
= 46986 mod 11327
= 1618
untuk proses berikutnya dapat dilakukan seperti diatas sampai dengan selesai. Hasil perhitungan dekripsi keseluruhannya dapat dilihat pada tabel berikut :

Tabel 2 Hasil Perhitungan *Enkripsi*

k[x]	V_n	Hasil
0	V_2	8630
1	V_3	1618
0	V_6	3236
1	V_7	7551

Dengan menggunakan cara yang sama untuk blok berikutnya, maka hasil akhir dari proses enkripsi akan didapatkan nilai 7551 dan 8158, langkah berikutnya adalah mengembalikan nilai tersebut kedalam karakter, sehingga hasil akhir setelah diubah dalam karakter adalah K3Q**:.**

3.1.3. Proses Dekripsi

Pada proses enkripsi sebelumnya didapatkan chiperteks yang berisi K3Q**:** dimana karakter-karakter tersebut akan dikembalikan menjadi teks seperti semula. Langkah awal proses dekripsi adalah membagi chiperteks menjadi blok-blok yang berisikan dua karakter, maka dari chiperteks yang dihasilkan pada proses enkripsi didapatkan K3 dan Q**:**. Selanjutnya tiap blok di konversi kedalam nilai ASCII dan didapatkan K3 = 7551 dan Q**:** = 8158. Untuk blok pertama didekripsi dengan menggunakan kunci privat yang telah dibangkitkan (103,11327) proses dekripsi dilakukan dengan menggunakan persamaan dekripsi $M = V_d(C_i \bmod N,1)$ Dimana *d* = 103, ci = nilai ASCII tiap blok, *N* = 11327, dan M adalah teks asli, misalkan *m*i adalah hasil dekripsi tiap blok, langkah berikutnya adalah membangkitkan *k*[*x*].

Tabel 3 Barisan Rantai Lucas Dekripsi

x	k[x]	D
1	1	d-1 = 102
2	0	d/2 = 51
3	1	d-1 = 50
4	0	d/2=25
5	1	d-1 = 24
6	0	d/2 = 12
7	0	d/2 = 6
8	0	d/2 = 3
9	1	d-1 = 2
10	0	d/2 = 1

Karena *k*[*x*] berfungsi sebagai Barisan Rantai Lucas, maka *k*[*x*] = {0, 1, 0, 0, 0, 1, 0, 1, 0, 1}, selanjutnya dilakukan proses dekripsi dengan menggunakan rumus yang telah ada tergantung pada nilai *k*[*x*]. Adapun proses perhitungan dekripsi adalah sebagai berikut :
Cipherteks : K3Q**:**
$M = V_d(c_i,1) \bmod N$
= 2 (15709,1) mod 11327
= 31418 mod 11327
= 8764
$M = V_d(c_i,1) \bmod N$
= 3 (15709,1) mod 11327
= 47127 mod 11327
= 1819

Untuk proses berikutnya dapat dilakukan seperti diatas sampai dengan selesai. Hasil perhitungan dekripsi keseluruhannya dapat dilihat pada tabel berikut:

Tabel 4 Hasil Perhitungan *Dekripsi*

k[x]	V_n	Hasil
1	V_2	8764
0	V_3	1819
1	V_6	3638
0	V_{12}	3805
1	V_{24}	2117
0	V_{25}	9736
0	V_{50}	5358
0	V_{51}	1403
1	V_{102}	8836
0	V_{103}	6869

Hasil akhir proses dekripsi untuk blok pertama adalah 6869, jika diubah dalam bentuk karakter maka hasilnya adalah DE. Blok-blok berikutnya mengikuti langkah yang sama seperti perhitungan diatas, sehingga memperoleh nilai 8773 dan diubah kedalam karakter menjadi DI, maka dapat dihasilkan teks asli M Kembali.

3.2. Algoritma Pembangkit Kunci

Algoritma Pembangkit Kunci digunakan dalam proses simulasi. Algoritma proses pembangkit kunci dalam bentuk *pseudocode* adalah sebagai berikut :

Input :
p ← bilangan prima;
q ← bilangan prima;
Output :
n ← Modulo;
e ← Kunci Publik;
d ← Kunci Privat;
Proses :
Φ(n) = (p-1) (p+1) (q-1) (q+1);
e = Φ(n);
d = m^{2-} 4.

3.3. Algoritma Proses *Enkripsi* LUC

Algoritma proses *enkripsi* LUC dalam bentuk *pseudocode* adalah sebagai berikut :

Input :
m_i ← Plaintext;
N ← Modulo;
Output :
V_e ← Barisan rantai lucas *enkripsi*;
Proses :
$C_i = V_e(m_i,1) \bmod N$;

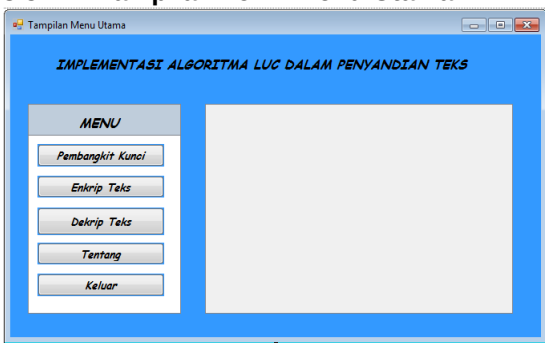
3.4. Algoritma Proses *Dekripsi* LUC

Algoritma proses *dekripsi* LUC dalam bentuk *pseudocode* adalah sebagai berikut :

Input :
C_i ← Chipertext;
N ← Modulo;
Output :
V_d ← Barisan rantai lucas *dekripsi*;
Proses :
$M = V_d(C_i, 1) \bmod N$;

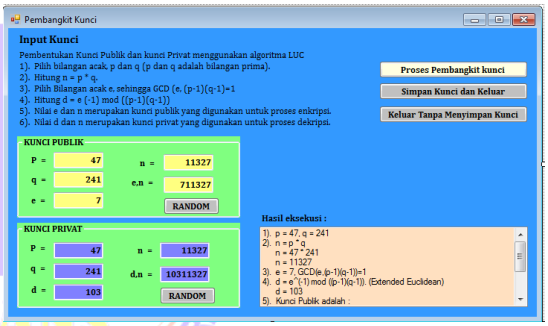
3.5. Implementasi Sistem

3.5.1. Tampilan *Form* Menu Utama



Gambar 1 Tampilan *Form* Menu Utama

3.5.2. Tampilan *Form* Pembangkit Kunci

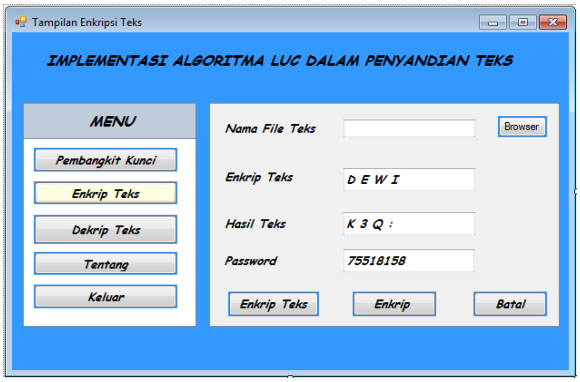


Gambar 2 Tampilan *Form* Pembangkit Kunci

3.5.3. Tampilan *Form Enkripsi* Teks

Gambar 3 Tampilan Form Enkripsi Teks

3.5.4. Tampilan *Form Dekripsi* Teks



Gambar 4 Tampilan Form Dekripsi Teks

KESIMPULAN

Berdasarkan hasil dan pembahasan dapat ditarik kesimpulan sebagai berikut :

1. Pada Algoritma LUC memiliki tahapan proses yakni pembangkit kunci (kunci *public* dan *privat*), *enkripsi* dan *dekripsi.* Algoritma LUC mampu melakukan proses *enkripsi* dan *dekripsi* dengan baik untuk huruf kapital.
2. Perangkat lunak Algoritma LUC ini dapat meningkatkan keamanan bagi informasi yang terkandung di dalamnya menjadi lebih baik.
3. Menyempurnakan aplikasi ini agar dapat melakukan enkripsi dan dekripsi dengan baik terutama untuk karakter kecil.
4. Diharapkan Perangkat lunak atau program aplikasi ini bisa dikembangkan lagi dengan menggunakan bahasa pemrograman yang lain seperti : C++ dan Matlab.
1. Perangkat lunak ini dapat dikembangkan dengan menambahkan algoritma kunci publik lainnya, seperti: metode Rabin dan ElGamal.
2. Untuk pengembangan perangkat lunak ini, sebaiknya dapat memproses *enkripsi* dan *dekripsi* pada semua jenis *file,* misalnya : gambar dan video.

DAFTAR PUSTAKA

[1]. Abdul Kadir, “Pengenalan Sistem Informasi”, Penerbit ANDI, Yogyakarta, 2003
[2]. Abdul Kadir dan Terra Ch. Triwahyuni, “Pengantar Teknologi Informasi Edisi Revisi”, Penerbit ANDI, Yogyakarta, 2013
[3]. C. Widyo Hermawan (ed), “Visual Basic 2008”, Penerbit ANDI, Yogyakarta, 2009
[4]. Dony Ariyus dan Rum Andri K.R, “Komunikasi Data”, Penerbit ANDI, Yogyakarta, 2008
[5]. Indrajani, “Database Design”, PT. Elex Media Komputindo, Jakarta, 2015
[6]. Rifki Sadikin, “Kriptografi untuk Keamanan Jaringan”, Penerbit ANDI, Yogyakarta, 2012
[7]. Rinaldi Munir, “Kriptografi”, Penerbit Informatika, Bandung, 2006
[8]. Ariwibowo, 2008. Aplikasi Pengamanan Dokumen Office Dengan Algoritma Kriptografi Kunci Asimetris Elgamal. Penelitian. Yogyakarta : Universitas Ahmad Dahlan Yogyakarta.
[9]. Riyanto, M.Z., 2007. Pengamanan Pesan Rahasia Menggunakan Algoritma Kriptografi Elgamal Atas Grup Pergandaan Zp*. Penelitian. Yogyakarta : Universitas Gajah Mada.
[10]. Saputra, R., Yismianto, B., dan Suhartono. 2006. Kriptografi Teks Dengan Menggunakan Algoritma LUC. Penelitian. Semarang : Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Diponegoro.
[11]. Nurli Hairiah, ISSN : 2339 – 210X, Volume V No. 1, Analisa dan Implementasi Algoritma Cipherblok Chaining Dalam Penyandian Teks, 2015.
[12]. http://eprints.undip.ac.id/99270/1/kriptografi_teks_dengan_menggunakan_algorit ma.luc.pdf 27 April 2015.