Implementasi Pengamanan Pesan Chatting menggunakan Metode Vigenere Cipher dan Cipher Block Chainning

Helmi Sahara
STMIK Budi Darma Medan, Jl. SM.Raja No.338 Sp.Limun Medan, Sumut, Indonesia
E-Mail : helmisahara23@gmail.com

ABSTRACT
Along with the increasingly rapid development of technology at this time, causing security is a very important thing. In this era, there are many terms of chatting. Chatting is a form of communication that is usually carried out between two people or more directly or realtime by utilizing network facilities. Various ways can be done by irresponsible parties to damage, hack our chat facilities, and therefore need security. To overcome this so that chat messages are safe from unauthorized users, we need a software that can do a message encryption / decryption process. In this paper we discuss the implementation of the Vigenere Cipher method and Chainning Cipher Block for the encryption / decryption process of securing chat messages.

Keywords: Vigenere Cipher, Chainning Cipher Block, Chat Message

PENDAHULUAN

Saat ini sistem komputer yang terpasang makin mudah diakses. Sistem *time sharing* dan akses jarak jauh menyebabkan masalah keamanan menjadi salah satu kelemahan komunikasi data seperti internet. Disamping itu kecenderungan lain saat ini adalah memberikan tanggung jawab sepenuhnya ke komputer untuk mengelola aktifitas pribadi dan bisnis seperti sistem transfer dana elekronis yang melewatkan uang sebagai aliran bit dan lain sebagainya. Untuk keamanan data, diperlukan kriptografi dengan metode enkripsi.

Kriptografi merupakan suatu seni dimana sebuah data diamankan melalui proses penyandian. Pada permulaannya kriptografi digunakan untuk mengamankan sebuah data berupa teks. Dan sekarang telah berkembang juga untuk pengamanan data berupa gambar. Sedangkan pembahasan ini penulis akan membahas tentang pengamanan data berupa teks [1][2].

Vigenere Cipher merupakan algoritma kriptografi klasik, sistem sandi poli-alfabetik yang sederhana. Sistem sandi poli-alfabetik mengenkripsi sekaligus sebuah teks yang terdiri dari beberapa huruf. Sandi *Vigenere* menggunakan subtitusi dengan fungsi *shift.* Berbeda dengan *Cipher Block Chainning* (CBC) sandi ini merupakan algoritma kriptografi modern, CBC menggunakan larik khusus yang disebut IV seukuran dengan ukuran blok (n bit)[3].

Berdasarkan latar belakang masalah diatas, maka rumusan masalah dalam penulisan skirpsi ini adalah sebagai berikut :

1. Bagaimana cara mengamankan pesan *chatting* dari pihak yang ingin merusak ?
2. Bagaimana proses enkripsi dan dekripsi pesan *chatting* menggunakan algoritma V*igenere Cipher* dan *Cipher Block Chainning* (CBC) ?
3. Bagaimana merancang aplikasi pengamanan pesan *chatting* menggunakan bahasa pemrograman?

Adapun batasan masalah dalam pembuatan penelitian ini adalah sebagai berikut:

1. Pengamanan hanya terhadap pesan teks.
2. Data yang di amankan akan di enkripsikan.
3. Metode yang akan di pakai dalam penyelesaian masalah pengamanan pesan *chatting* adalah algoritma V*igenere Cipher* dan *Cipher Block Chainning* (CBC).
4. Untuk merancang program pengamanan pesan *chatting* akan di buat dengan menggunakan bahasa pemograman Visual Basic. Net 2008

LANDASAN TEORI

2.1 Kriptografi

Kriptografi merupakan seni dan ilmu menyembunyikan informasi dari penerima yang tidak berhak. Kata *cryptography* berasal dari kata Yunani *kriptos* (tersembunyi) dan *graphein* (menulis)[4][5]. Kriptografi pada awalnya dijabarkan sebagai ilmu yang mempelajari bagaimana menyembunyikan pesan. Namun pada pengertian modern kriptografi adalah ilmu yang bersandarkan pada teknik matematika untuk berurusan dengan keamanan informasi seperti kerahasiaan, keutuhan data dan otentikasi entitas.

2.2 Algoritma *Vegenere Cipher*

Vegenere Cipher merupakan algoritma kriptografi klasik, sistem sandi poli-alfabetik yang sederhana. Sistem sandi poli-alfabetik mengenkripsi sekaligus sebuah teks yang terdiri dari beberapa huruf. Sandi *Vegenere* menggunakan subtitusi dengan fungsi *shift*[4][1]. Kelas Vigenere Cipher memiliki 2 fungsi enkripsi dan dekripsi. Pada fungsi enkripsi tiap karakter ke-i pada teks asli ditambah dengan kunci indeks ke-i mod m dengan m adalah panjang kunci Vigenere. Sedangkan pada fungsi dekripsi tiap karakter ke-i pada teks sandi dikurang dengan kunci indeks ke-i mod m.

Rumus : Ci = (P1 + K1) Mod 256

Dimana : C = cipherteks yang akan di cari dari proses enkripsi

P = plainteks awal yang akan di enkripsi

K = kunci

Contoh sederhana proses enkripsi vigenere cipher :

Plainteks = HEL				Kunci = STM		
H	E	L		S	T	M
72	69	76		83	84	77

C1 = (P1 + K1) Mod 256 C2= (P2 + K2) Mod 256 C3 = (P3 + K3) Mod 256

= (H + S) Mod 256 = (E + T) Mod 256 = (L + M) Mod 256

= (72 + 83) Mod 256 = (69 + 84) Mod 256 = (76 + 77) Mod 256

= 155 Mod 256 = 153 Mod 256 = 153 Mod 256

= 155 ↔ > = 153 ↔ ™ = 153 ↔ ™

2.3 Algoritma *Cipher Block Chainning* (CBC)

CBC merupakan algoritma kriptografi modern, CBC menggunakan larik khusus yang disebut IV seukuran dengan ukuran blok (n bit). CBC direkomendasikan digunakan dengan sistem sandi yang memiliki ukuran blok lebih dari 64 bit. Operasi dapat dilakukan untuk mewujudkan layanan otentikasi. Pada proses enkripsi CBC, plainteksnya adalah hasil cipherteks pada operasi vigenere cipher sebelumnya, dimana setiap karakter akan dibuat nilai desimalnya, dan pada nilai desimalnya akan di ubah menjadi nilai biner untuk melanjutkan proses enkripsi CBC. Sebelumnya CBC memiliki ketentuan dimana CBC memiliki C_0 dan nilai C_0 dapat kita tentukan sendiri. C_0 berfungsi sebagai nilai awal pada blok yang pertama di dalam proses XOR enkripsi dan dekripsi nantinya. Sedangkan blok yang ke dua akan XOR dengan hasil blok-blok sebelumnya.

PEMBAHASAN

3.1 Analisa Masalah

Di dalam pesatnya perkembangan teknologi sekarang ini maka semakin banyak pula macam serangan terhadap teknologi tersebut. Salah satunya adalah serangan terhadap data atau file yang ada di jaringan komputer yang di rusak, di modifikasi, atau bahkan di hancurkan oleh orang-orang yang tidak bertanggungjawab[6]. Media *internet* memungkinkan pengguna untuk berkomunikasi secara *real time* dengan menggunakan antarmuka *web,* seperti *chatting, email* dan lain sebagainya yang mudah diakses bagi pengguna untuk berkomunikasi dengan pesan teks. Jika pesan teks pada *chatting* sangat penting dan harus dijaga kerahasiaannya maka diperlukan pengamanan pesan teks pada media *chatting* yang akan dikirim dengan menggunakan algoritma kriptografi untuk mencegah dari pihak yang ingin merusak. Melihat hal tersebut banyak pula kini diciptakan cara-cara untuk mengamankan [3].

Berdasarkan analisa di atas maka penulis ingin menggunakan algoritma *Vigenere Cipher* dan *Cipher Block Chainning* (CBC) untuk mengamankan pesan sms dari orang-orang yang hendak merusaknya. *Vigenere Cipher* dan *Cipher Block Chainning* (CBC) juga akan melakukan proses enkripsi dan dekripsi, cipherteks *Vigenere Cipher* akan dilanjutkan proses enkripsi CBC, begitu pula sebaliknya dalam proses dekripsi. Dalam pembahasan ini penulis menggunakan dua metode pengamanan, diharapkan agar dapat lebih mempersulit proses perusakan oleh pihak yang tidak bertanggungjawab.

3.2 Penerapan Algoritma *Vigenere Cipher* dan *Cipher Block Chainning* (CBC)

Berikut adalah penerapan algoritma *vigenere cipher* dan CBC dalam mengamankan pesan *chatting*. Disesuaikan dengan aplikasi yang akan dibangun dengan pemakaian algoritma *Vigenere Cipher* dan *Block Cipher Chainning* (CBC).

ENKRIPSI :

Proses enkripsi algoritma *Vigenere Cipher.*

Plainteks = H E L M I _ S A H A R A _ A L

H	E	L	M	I	_	S	A	H	A	R	A	_	A	L
72	69	76	77	73	95	83	65	72	65	82	65	95	65	76

Kunci = S T M I K _ B U D I D A R M A

S	T	M	I	K	_	B	U	D	I	D	A	R	M	A
83	84	77	73	75	95	66	85	68	73	68	65	82	77	65

Ci = (P1 + K1) Mod 256

Keterangan :
C = cipherteks yang akan di cari dari proses enkripsi
P = plainteks awal yang akan di enkripsi
K = kunci
Catatan :
Sebelum melakukan proses enkripsi baiknya perhatikan langkah-langkah berikut ini:

1. Jumlah karakter kunci harus sama dengan jumlah karakter plainteks. Jika kurang, lakukan pengurangan penulisan kunci hingga jumlah karakternya sama dengan jumlah plainteks.
2. Konversi setiap karakter plainteks dan kunci ke desimal.
3. Konversi setiap nilai cipherteks ke *Char.*

C1 = (P1 + K1) Mod 256
= (H + S) Mod 256
= (72 + 83) Mod 256
= 155 Mod 256
= 155 ↔ >

C2 = (P2 + K2) Mod 256
= (E + T) Mod 256
= (69 + 84) Mod 256
= 153 Mod 256
= 153 ↔ ™

C3 = (P3 + K3) Mod 256
= (L + M) Mod 256
= (76 + 77) Mod 256
= 153 Mod 256
= 153 ↔ ™

C4 = (P4 + K4) Mod 256
= (M + I) Mod 256
= (77 + 73) Mod 256
= 150 Mod 256
= 150 ↔ -

C5 = (P5 + K5) Mod 256
= (I + K) Mod 256
= (73 + 75) Mod 256
= 148 Mod 256
= 148 ↔ ”

C6 = (P6 + K6) Mod 256
= (_ + _) Mod 256
= (95 + 95) Mod 256
= 190 Mod 256
= 190 ↔ ¾

C7 = (P7 + K7) Mod 256
= (S + B) Mod 256
= (83 + 66) Mod 256
= 149 Mod 256
= 149 ↔ •

C8 = (P8 + K8) Mod 256
= (A + U) Mod 256
= (65 + 85) Mod 256
= 150 Mod 256
= 150 ↔ -

C9 = (P9 + K9) Mod 256
= (H + D) Mod 256
= (72 + 68) Mod 256
= 140 Mod 256
= 140 ↔ Œ

C10 = (P10 + K10) Mod 256
= (A + I) Mod 256
= (65 + 73) Mod 256
= 138 Mod 256
= 138 ↔ Š

C11 = (P11 + K11) Mod 256
= (R + D) Mod 256
= (82 + 68) Mod 256
= 150 Mod 256
= 150 ↔ -

C12 = (P12 + K12) Mod 256
= (A + A) Mod 256
= (65 + 65) Mod 256
= 130 Mod 256
= 130 ↔ ,

C13 = (P13 + K13) Mod 256
= (_ + R) Mod 256
= (95 + 82) Mod 256
= 177 Mod 256
= 177 ↔ ±

C14 = (P14 + K14) Mod 256
= (A + M) Mod 256
= (65 + 77) Mod 256
= 142 Mod 256
= 142 ↔ Ž

C15 = (P15 + K15) Mod 256
= (L + A) Mod 256
= (76 + 65) Mod 256
= 141 Mod 256
= 141 ↔ 141

Cipherteks Vigenere Cipher :
Plainteks CBC :

>	™	™		155	153	153
-	”	¾		150	148	190
•	-	Œ		149	150	140
Š	-	,		138	150	130
±	Ž	141		177	142	141

NB :
Didalam proses enkripsi CBC terlebih dahulu menentukan panjang bit / kelompok, disini sebagai lanjutan memakai 3 x 8 bit = 24 bit.
Jika pada plainteks dari hasil cipherteks *vigenere* belum cukup, maka dilakukan penambahan karakter sesuai yang di butuhkan. Dan pada enkripsi CBC harus ditentukan nilai C_0 dan kunci.

Karakter Tambahan						
<	ƒ	„		139	131	132
…	†	‡		133	134	135
^	‰	143		136	137	143

C_0 = B _ D

C_0	B	66	1000010
	_	95	1011111
	D	68	1000100

K = U A S

K	U	85	1010101
	A	65	1000001
	S	83	1010011

ENKRIPSI CBC

Proses konversi nilai karakter ke biner :

P1 = > = 155 = 10011011
= ™ = 153 = 10011001
= ™ = 153 = 10011001

P2 = - = 150 = 10010110
= ” = 148 = 10010100
= ¾ = 190 = 10111110

P3 = • = 149 = 10010101
= - = 150 = 10010110
= Œ = 140 = 10001100

P4 = Š = 138 = 10001010
= - = 150 = 10010110
= , = 130 = 10000010

P5 = ± = 177 = 10110001
= Ž = 142 = 10001110
= 141 = 141 = 10001101

P6 = < = 139 = 10001011
= ƒ = 131 = 10000011
= „ = 132 = 10000100

P7 = … = 133 = 10000101
= † = 134 = 10000110
= ‡ = 135 = 10000111

P8 = ^ = 136 = 10001000
= ‰ = 137 = 10001001
= 143 = 143 = 10001111

Cipherteks CBC :

110010000111100011101000
101110101101000001010000
101000000111100011110111
111110101111001001100111
111000111101101110010001
110100011001010001100011
000101010011101101110000
100011110011101011001100

Sebelum melakukan Dekripsi ubah dahulu semua posisi bit yang sebelumnya di Shiftkan, untuk kembali seperti semula.

Cp1 = 110010000111100011101000 ↔ 100011001000011110001110 (Cp1)
Cp2 = 101110101101000001010000 ↔ 000010111010110100000101 (Cp2)
Cp3 = 101000000111100011110111 ↔ 011110100000011110001111 (Cp3)
Cp4 = 111110101111001001100111 ↔ 011111111010111100100110 (Cp4)
Cp5 = 111000111101101110010001 ↔ 000111100011110110111001 (Cp5)
Cp6 = 110100011001010001100011 ↔ 001111010001100101000110 (Cp6)
Cp7 = 000101010011101101110000 ↔ 000000010101001110110111 (Cp7)
Cp8 = 100011110011101011001100 ↔ 110010001111001110101100 (Cp8)

Cipherteks dekripsi CBC :
Plainteks Dekripsi *Vigenere Cipher* :

P	155	153	153	150	148	190	149	150	140	138	150	130	177	142	141
K	83	84	77	73	75	95	66	85	68	73	68	65	82	77	65

NB :
Plainteks dekripsi *Vigenere Cipher* telah dilakukan pengurangan, untuk menyamakan jumlah kunci pada *vigenere* dengan plainteksnya.

DEKRIPSI *VIGENERE CIPHER* :

P1 = (C1 – K1) Mod 256
= (155 – 83) Mod 256
= 72 ↔ H

P2 = (C2 – K2) Mod 256
= (153 – 84) Mod 256
= 69 ↔ E

P3 = (C3 – K3) Mod 256
= (153 – 77) Mod 256
= 76 ↔ L

P4 = (C4 – K4) Mod 256
= (150 – 73) Mod 256
= 77 ↔ M

P5 = (C5 – K5) Mod 256
= (148 – 75) Mod 256
= 73 ↔ I

P6 = (C6 – K6) Mod 256
= (190 – 95) Mod 256
= 95 ↔ _

P7 = (C7 – K7) Mod 256
= (149 – 66) Mod 256
= 83 ↔ S

P8 = (C8 – K8) Mod 256
= (150 – 85) Mod 256
= 65 ↔ A

P9 = (C9 – K9) Mod 256
= (140 – 68) Mod 256
= 72 ↔ H

P10 = (C10 – K10) Mod 256
= (138 – 73) Mod 256
= 65 ↔ A

P11 = (C11 – K11) Mod 256
= (150 – 68) Mod 256
= 82 ↔ R
P12 = (C12 – K12) Mod 256
= (130 – 65) Mod 256
= 65 ↔ A
P13 = (C13 – K13) Mod 256
= (177 – 82) Mod 256
= 95 ↔ _
P14 = (C14 – K14) Mod 256
= (142 – 77) Mod 256
= 65 ↔ A
P15 = (C15 – K15) Mod 256
= (141 – 65) Mod 256
= 76 ↔ L

Cipherteks *Vigenere Cipher* :
H E L M I _ S A H A R A _ A L

ALGORITMA DAN IMPELEMENTASI

Algoritma merupakan urutan langkah-langkah logis dalam penyelesaian masalah yang disusun secara sistematis. Langkah-langkah yang tidak benar dapat memberikan hasil yang salah. Pada algoritma pembelajaran ini akan melakukan tahapan-tahapan perancangan aplikasi *chatting* kriptografi Algoritma *Vigenere Cipher* dan algoritma *Chiper Block Chaining (CBC).*

4.1 Algoritma Enkripsi Halaman *Chatting*

Pada algoritma halaman *chatting* ini menjelaskan bagaimana tahapan algoritma *Vigenere Cipher* dan algoritma *Chiper Block Chaining (CBC).* melakukan proses enkripsi. Prosedur kerja dari algoritma enkripsi dapat dijabarkan sebagai berikut :

Input : *Username(U), Plainteks(*P*),* kunci(k)
Output : *View* halaman *Chatting, chipher*teks*,*
Proses :

Deklarasi :
Username, kunci, newKey,
plaintext : string;
karakter:array[1 …..
karakter.length]of string;
i, j : integer;
If(u←”kosong”) then
View *username* masih kosong
Else
View halaman *chatting*
End if
If(P and k ← “kosong”)
View plainteks dan kunci masih kosong
Else
for (int i ← 0; i < plaintext.length)do
j ← i mod key.length();
newKey ← kunci(j);
end for;
write(newKey);
karakter [i] ← plaintext;
kunci ← get(key.plaintext);
for (int i ← 0; i < karakter.length)do
ciphertext←(char) (((256+((karakter[i]-‘ ’)+key(i)-‘ ’)) mod 256+’ ’);
end for
write (ciphertext);
Prosedur algoritma CBC{
konversi *ciperteks vigenere menjadi plainteks CBC* (P) dan kunci (K) yang di-*input* ke dalam bentuk biner.
Kelompokkan *bit* biner plainteks (P) dan kunci (K) dengan dengan jumlah *bit* setiap kelompok sama }
P ← *n*-bit plainteks blok *M* = $M_1M_2 \ldots M_t$.
$C_j = E_K(C_{j-1}$ *XOR* $M_j)$
C← *n*-bit *cipher*teks blok *C* = $C_0C_1 \ldots C_t$
*Cipher*teks ← bin2dec
View *cipher*teks pada halaman obrolan *chatting*
End if

4.2 Algoritma Proses Dekripsi *Chatting*

Pada algoritma halaman *chatting* ini menjelaskan bagaimana tahapan algoritma *Vigenere Cipher* dan algoritma *Chiper Block Chaining (CBC).* melakukan proses dekripsi. dapat dijabarkan sebagai berikut :

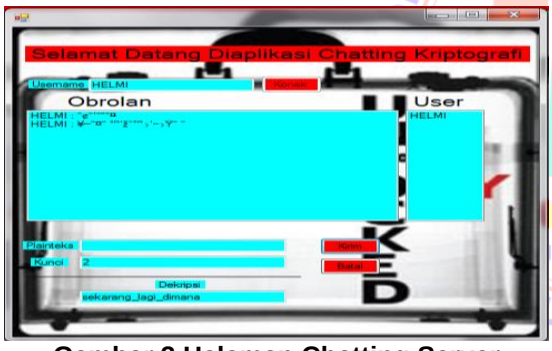
Input : cipher, kunci
Output : plainteks
Proses :
Deklarasi :
kunci, newKey, ciphertext : string;
karakter:array[1 ….. karakter.length]of string;
i, j : integer;
Prosedur algoritma CBC{
konversi *cipher*teks (C) dan kunci (K) yang di-*input* ke dalam bentuk biner.
Kelompokkan *bit* biner *cipher*teks (C) dan kunci (K) dengan dengan jumlah *bit* setiap kelompok sama }
C ← *n-bit cipher*teks blok *C* = $C_0C_1 \ldots C_t$.
$M_j = C_{j-1}$ *XOR* $D_K(C_j)$.
P← *n*-bit *plain*teks blok *M* = $M_1M_2 \ldots M_t$.
*Plain*teks ← bin2dec
Prosedur algoritma *Vigenere*{
Asumsikan plainteks algoritma CBC menjadi *cipher*teks algoritma vigenere
}
for (int i ← 0; i < ciphertext.length)do
j ← i mod key.length();
newKey ← key.charAt(j);
end for;
write(newKey);
karakter [i] ← ciphertext;

```
            key ← get(key. ciphertext);
    for (int i ← 0; i < karakter.length)do
            plaintext ← (char)
            (((256+((karakter[i] – ‘ ’)-
            key.charAt(i) – ‘ ‘)) mod
            256)+’ ’);
    end for
    write (plaintext);
```

4.3 Implementasi

Perancangan aplikasi kriptografi pesan *chatting* telah dirancang dan dibuat dengan menggunakan aplikasi Microsoft Visual Studio 8 dan bahasa pemograman Visual Basic.

Gambar 2 Tampilan Halaman Menu Chatting



Gambar 3 Halaman Chatting Server

KESIMPULAN

Sebagai penutup sajian pembahasan dalam penulisan penelitian ini penulis mencoba mengambil kesimpulan-kesimpulan sekaligus memberikan saran.

Dari pembahasan dari bab-bab sebelumnya, maka penulis menarik kesimpulan sebagai berikut :

1. Pentingnya melakukan pengamanan pesan *chatting,* agar terhindar dari pihak-pihak yang ingin merusak.
2. Salah satu cara mengamankan pesan *chatting* dapat dilakukan dengan menggunakan metode V*igenere Cipher* dan *Cipher Block Chainning* (CBC) dengan proses enkripsi dan dekripsi.
3. Pengamanan pesan *chatting* dapat dilakukan dengan merancang sistem aplikasi menggunakan bahasa pemograman.

DAFTAR PUSTAKA

[1] D. Ariyus, *Pengantar Ilmu Kriptografi: Teori Analisis & Implementasi*. Penerbit Andi.

[2] T. Limbong and P. D. P. Silitonga, “Testing the Classic Caesar Cipher Cryptography using of Matlab,” *Int. J. Eng. Res. Technol.*, vol. 6, no. 2, pp. 175–178, 2017.

[3] D. Lombu, S. D. Tarihoran, and I. Gulo, “Kombinasi Mode Cipher Block Chaining Dengan Algoritma Triangle Chain Cipher Pada Penyandian Login Website,” *J-SAKTI (Jurnal Sains Komput. dan Inform.*, vol. 2, no. 1, pp. 1–11, 2018.

[4] R. Munir, “Kriptografi,” *Inform. Bandung*, 2006.

[5] T. Limbong *et al.*, “The implementation of computer based instruction model on Gost Algorithm Cryptography Learning,” in *IOP Conference Series: Materials Science and Engineering*, 2018, vol. 420, no. 1, p. 12094.

[6] J. Simarmata, “Pengamanan Sistem Komputer,” *Andi, Yogyakarta*, 2006.