Rancangan Algoritma Shift Vigenere Cipher

Arizka Dikson 13504133
Program Studi Teknik Informatika ITB, Bandung, email: if14133@students.if.itb.ac.id

Abstract – *Makalah ini membahas tentang rancangan dan implementasi algoritma kriptografi Shift Vigenere Cipher. Algoritma ini merupakan hasil rekayasa dari algoritma Vigenere dengan memanfaatkan panjang kunci. Algoritma Shift Vigenere yang akan dirancang ini akan selalu menghasilkan panjang kunci dalam proses enkripsi yang lebih panjang dibandingkan kunciyang diingat oleh user .*

Makalah ini membahas konsep dasar algoritma Shift Vigenere, proses enkripsi dan dekripsinya, dan hasil implementasi sederhana algoritma tersebut menggunakan bahasa pemrograman C# dan memanfaatkan kakas Visual Studio 2005. Selain itu, makalah juga membahas tingkat keamanan algoritma ini ditinjau dari aspek kelebihan dan kekurangan algoritma, serangan kriptanalisis yang mungkin, serta arah pengembangan algoritma ini ke depannya.

Kata Kunci*: algoritma klasik, cipher, vigenere, shift, enkripsi, dekripsi*

1. PENDAHULUAN

1.1. Kriptografi

Algoritma kriptografi (*cipher*) adalah aturan untuk *enciphering* dan *deciphering* atau fungsi matematika yang digunakan untuk enkripsi dan dekripsi. Berdasarkan sejarah algoritma kriptografi dapat dibedakan menjadi algoritma kriptografi klasik dan algoritma kriptografi modern. Algoritma yang akan dirancang berikut ini termasuk dalam algoritma klasik.

1.2. Algoritma Klasik

Algoritma kriptografi klasik yaitu algoritma kriptografi sebelum masuk era digital, Kriptografi yang dilakukan berbasis karakter. Algoritma kriptografi klasik termasuk dalam sistem kriptografi simetri, karena kunci untuk melakukan enkripsi sama dengan kunci untuk melakukan enkripsi.

1.3. *Chiper* Subtitusi

Algoritma kriptografi substitusi merupakan cipher tertua. Prinsip utama cipher substitusi adalah menukarkan setiap huruf pada *plaintext* dengan huruf lain/simbol.

2. RANCANGAN ALGORITMA SHIFT VIGENERE

2.1. Latar Belakang

Algoritma vigenere cipher merupakan salah satu jenis algoritma kriptografi cipher klasik yang sudah digunakan sejak pecahnya perang sipil Amerika. Prinsip dasar algoritma ini adalah dengan mengenkripsikan setiap karakter pada plainteks dengan kunci menggunakan tabel vigenere. Pada algoritma ini terjadi pengulangan kunci setelah proses enkripsi sejumlah panjang kunci, sehingga keamanan algoritma ini sangat bergantung sekali pada panjang kunci yang digunakan.

Apabila seseorang ingin mengirimkan pesan yang dienkripsi menggunakan metode vigenere cipher, akan tetapi dia ingin kunci yang digunakan merupakan kata yang gampang diingat dan keamanan pesannya juga sama seperti kunci yang panjang, maka hal ini tentunya tidak dapat dipecahkan dengan algoritma vigenere yang biasa saja. Sisi inilah yang ingin dieksplorasi penulis dengan metode shift vigenere cipher yang berfungsi untuk mengenkripsi plainteks dengan metode vigenere akan tetapi dengan menambahkan sebuah fungsi untuk memodifikasi kunci sehingga pada proses enkripsi n karakter berikutnya kunci tidak akan berulang, dimana n = panjang kunci.

2.2. Shift Vigenere Cipher

Algoritma ini terinspirasi dari salah satu metode algoritma kriptografi modern yakni dalam hal *keystream generator* dan *shift-bit.* Jika dalam algoritma kriptografi modern keystream generator digunakan untuk membangkitkan bit-bit kunci untuk di-xor kan dengan bit-bit plainteks, maka dalam algoritma shift vigenere ini, keystream generatornya merupakan proses modifikasi kunci untuk menghasilkan karakter-karakter kunci yang akan dienkripsi dengan karakter pada plainteks yang berkoresponden.

Pada awalnya, penulis ingin mengembangkan algoritma ini sehingga keystream generator akan menghasilkan aliran kunci yang panjangnya sama dengan plainteks sehingga kemungkinan keberhasilan para kriptanalisis untuk menyerang dengan metode Kasiski akan sangat kecil dan salah satu syarat unbreakable cipher akan terpenuhi. Akan tetapi karena berbagai kendala yang dihadapi penulis, maka untuk saat ini metode ini hanya bisa memperpanjang kunci dengan jumlah tertentu saja.

2.3. Proses Enkripsi

Proses enkripsi plainteks menjadi cipherteks sama seperti vigenere cipher dengan sedikit modifikasi pada kunci. Langkah-langkah pengenkripsian dapat dilihat sebagai berikut:

a. Dekripsikan karakter pertama pada plainteks dengan karakter pertama pada kunci dengan mengikuti kaidah tabel vigenere seperti pada vigenere cipher biasa.
b. Untuk karakter kedua hingga karakter ke-*n* dimana n=panjang kunci, ulangi langkah a.
c. Setelah selesai satu periode pengenkripsian, bentuk kunci baru dari kunci lama dengan menggeser satu karakter ke kiri.
d. Enkripsikan karakter-karakter berikutnya pada plainteks dengan mengulangi langkah a, b, c secara berurutan terus-menerus hingga karakter terakhir pada plainteks selesai dienkripsi.

Untuk lebih jelasnya dapat dilihat pada contoh di bawah ini dimana plainteks yang ingin dienkripsi adalah "THIS PLAINTEKS" dan kunci yang digunakan adalah "SONY".

Plainteks	`THIS PLAINTEXT`
Kunci	`SONY ONYSNYSOY`
Chiperteks	**`LVVQ DYYAARWLR`**

Pada contoh di atas dapat dilihat pergesaran kunci yang terjadi sebanyak 1 karakter ke kiri setelah 4 karakter sehingga proses pergesaran kunci dapat dilihat sebagai berikut:

`SONY` -> **`ONYS`** -> **`NYSO`** -> **`Y`**`SON`

Dari gambaran di atas, maka dapat dilihat bahwa proses enkripsi dengan panjang kunci = 4, pengulangan enkripsi dengan kunci yang pola yang sama akan terjadi setelah 16 karakter. Dari contoh tersebut maka dapat disimpulkan bahwa pengulangan kunci akan terjadi setelah n^2 dimana n = panjang kunci awal.

2.3. Proses Dekripsi

Proses dekripsi cipherteks menjadi plainteks juga sama seperti vigenere cipher dengan sedikit modifikasi pada kunci. Langkah-langkah dalam proses dekripsi dapat dilihat sebagai berikut:

a. Dekripsikan karakter pertama pada cipherteks dengan karakter pertama pada kunci dengan mengikuti kaidah tabel vigenere seperti pada vigenere cipher biasa.
b. Untuk karakter kedua hingga karakter ke-*n* dimana n=panjang kunci, ulangi langkah a.
c. Setelah selesai satu periode pendekripsian, bentuk kunci baru dari kunci lama dengan menggeser satu karakter ke kiri.
d. Dekripsikan karakter-karakter berikutnya pada cipherteks dengan mengulangi langkah a, b, c secara berurutan terus-menerus hingga karakter terakhir pada cipherteks selesai didekripsi.

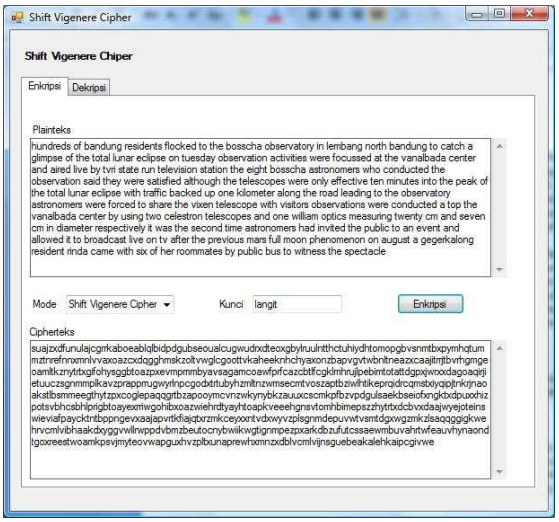
Untuk lebih jelasnya dapat dilihat pada contoh di bawah ini yang mendekripsi hasil enkripsi dari contoh enkripsi sebelumnya dengan kunci "SONY".

Chiperteks	`LVVQ DYYAARWLR`
Kunci	`SONY ONYSNYSOY`
Plainteks	**`THIS PLAINTEXT`**

3. HASIL DAN PEMBAHASAN

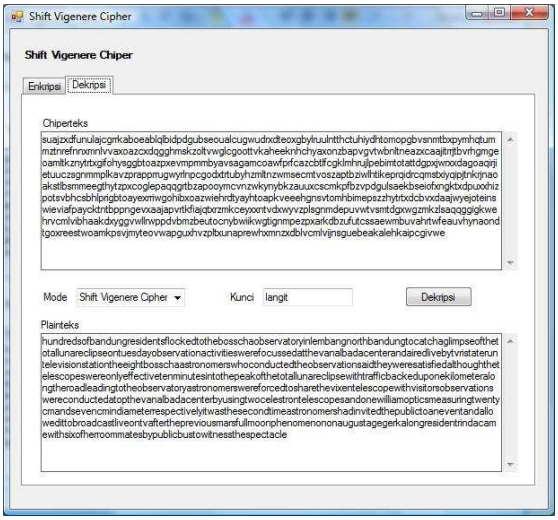
3.1. Implementasi

Rancangan algoritma shift vigenere ini sudah diimplementasikan penulis dengan menggunakan bahasa pemrograman C# dan kakas Microsoft Visual Studio 2005. Berikut adalah antarmuka aplikasinya:



Gambar 1 Halaman Enkripsi

Pada halaman enkripsi di atas, terdapat *textarea* untuk menuliskan plainteks, *combo box* untuk memilih metode enkripsi yang digunakan (Shift Vigenere Cipher dan Vigenere Cipher) dan *textbox* untuk mengisi kunci. Untuk melakukan enkripsi, klik tombol "Enkripsi", maka cipherteks hasil enkripsi akan muncul di *textarea* paling bawah.



Gambar 2 Halaman Dekripsi

Pada halaman dekripsi di atas, terdapat *textarea* untuk menuliskan cipherteks, *combo box* untuk memilih metode dekripsi yang digunakan (Shift Vigenere Cipher dan Vigenere Cipher) dan *textbox* untuk mengisi kunci. Untuk melakukan dekripsi, klik tombol "Dekripsi", maka plainteks hasil dekripsi akan muncul di *textarea* paling bawah.
Berikut adalah kode enkripsi dan dekripsi yang digunakan dalam implementasi algoritma ini:

```
public String shitLeft(Char[] kunci)
{
String hasil = "";
Char cc =kunci[0];
for (int i=0;i<kunci.Length; i++)
{if (i == kunci.Length - 1)
{ kunci[i] = cc; }
else
{kunci[i] = kunci[i + 1];}
hasil = hasil + kunci[i].ToString();
}
return hasil;
}

public String konversiKunciShift(int
length, String kunci)
{
String hasil = "";
int j = 0;
for (int i = 0; i < length; i++)
{if (j == kunci.Length)
{kunci = shitLeft(kunci.ToCharArray());
j = 0;}
hasil = hasil+kunci[j].ToString();
j++;}
return hasil;
}

public String
enkripsiStringShiftVigenere(String
plainteks, String kunci)
{
String plainTrim = plainteks.Replace(" ",
"");
Char[] cipher = new
Char[plainTrim.Length];
String cipherteks = "";
String kunciArray =
konversiKunciShift(plainTrim.Length,
kunci);
for (int i = 0; i < plainTrim.Length; i++)
{                cipher[i] =
enkripsiKar(plainTrim[i], kunciArray[i]);
                cipherteks = cipherteks +
cipher[i].ToString();}
return cipherteks;
}

public String
dekripsiStringShiftVigenere(String
cipherteks, String kunci)
{
String cipher = cipherteks.Replace("
","").ToLower();
String kunciArray =
konversiKunciShift(cipher.Length, kunci);
Char[] plain = new Char[cipher.Length];
String plainteks = "";
for (int i = 0; i < cipher.Length; i++)
{
plain[i] = dekripsiKar(cipher[i],
kunciArray[i]);
plainteks = plainteks +
plain[i].ToString();
}
return plainteks;
}
```

3.1. Percobaan Serangan Kriptanalisis

3.1.1. Deskripsi Persoalan

Persoalan yang diambil dalam mencoba serangan kriptanalisis pada algoritma shift vigenere ini adalah

sebuah teks sebagai berikut:

```
hundreds of bandung residents
flocked to the bosscha observatory
in lembang north bandung to catch a
glimpse of the total lunar eclipse
on tuesday observation activities
were focussed at the vanalbada
center and aired live by tvri state
run television station the eight
bosscha astronomers who conducted
the observation said they were
satisfied although the telescopes
were only effective ten minutes into
the peak of the total lunar eclipse
with traffic backed up one kilometer
along the road leading to the
observatory astronomers were forced
to share the vixen telescope with
visitors observations were conducted
a top the vanalbada center by using
two celestron telescopes and one
william optics measuring twenty cm
and seven cm in diameter
respectively it was thesecond time
astronomers had invited the public
to an event and allowed it to
broadcast live on tv after the
previous mars fullmoon phenomenon on
august a gegerkalong resident rinda
came with six of her roommates by
public bus to witness the spectacle
```

Plainteks di atas dienkripsi dengan menggunakan algoritma shift vigenere dengan kunci “langit”, maka akan dihasilkan cipherteks sebagai berikut:

```
suajzxdfunulajcgrrkaboeablqlbidpdgub
seoualcugwudrxdteoxgbylruulntthctuhi
ydhtomopgbvsnmtbxpymhqtummztnrefnnxm
nlvvaxoazcxdqgghmskzoltvwglcgoottvka
heeknhchyaxonzbapvgvtwbnltneazxcaaji
trrjtbvrhgmgeoamltkznytrtxgifohysggb
toazpxevmpmmbyavsagamcoawfprfcazcbtl
fcgklmhrujlpebimtotattdgpxjwrxxdagoa
qirjietuuczsgnmmplkavzprapprrugwyrln
pcgodxtrtubyhzmltnzwmsecmtvoszaptbzi
wlhtikeprqidrcqmstxiyqipjtnkrjnaoaks
tlbsmmeegthytzpxcoglepaqqgrtbzapooym
cvnzwkynybkzauuxcscmkpfbzvpdgulsaekb
seiofxngktxdpuxxhizpotsvbhcsbhlprigb
toayexrriwgohibxoazwiehrdtyayhtoapkv
eeehgnsvtomhbimepszzhytrtxdcbvxdaajw
yejoteinswieviafpaycktntbppngevxaaja
pvrtkfiajqtxrzmkceyxxntvdxwyvzplsgnm
depuvwtvsmtdgxwgzmkzlsaqqggigkwehrvc
mlvibhaakdxyggvwllrwppdvbmzbeutocnyb
wiikwgtignmpezpxarkdbzufutcssaewmbuv
ahrtwfeauvhynaondtgoxreestwoamkpsvjm
yteovwapguxhvzpltxunaprewhxmnzxdblvc
mlvijnsguebeakalehkaipcgivwe
```

Dalam percobaan kriptanalisis ini, penulis akan menggunakan metode Kasiski, yakni metode untuk mencari panjang kunci, kemudian setelah panjang kunci diperoleh, maka akan dilakukan analisis frekwensi dengan panjang kunci yang telah diperoleh tadi.

3.1.2. Langkah Kriptanalisis

Langkah pertama yang digunakan adalah menggunakan metode Kasiski untuk menentukan panjang kunci. Untuk kasus ini penulis mencari pasangan huruf dengan jumlah karakter > 4. Dari hasil percobaan yang dilakukan, maka diperoleh data sebagai berikut:

Tabel 1 Jarak antar Frase yang Sama

Pasangan Huruf(Frase)	Jarak Antar Frase
YTRTX	360
GBTOA	288
VCMLVI	144

Dari Tabel 1 diperoleh:

Tabel 2 Faktor Pembagi

Bilangan	Faktor Pembagi
360	{2,4,5,6,8,9,10,12,15,18,20,24,30,36,40, 45,60,72,90,180}
288	{2,4,6,8,9,12,16,18,24,36,72,144}
144	{2,4,6,8,9,12,16,18,24,36,72}

Sehingga diperoleh faktor pembagi bersama = {2,4,6,8,9,12,18,24,36,72}. Dengan mengabaikan {2}, maka kemungkinan panjang kunci ada sebanyak 9 kemungkinan. Jika pada suatu waktu kriptanalisis memilih panjang kunci = 36, maka langkah selanjutnya adalah dengan menganalisis frekwensi kemunculan pasangan huruf. Pasangan huruf yang paling bayak muncul adalah ‘GNM’, ‘TOA’, ‘TRT’, ‘AAJ’, ‘ZPX’ dan ‘OAZ’. Dengan melakukan percobaan mensubtitusi satu per satu pasangan huruf tadi dengan kata yang paling sering muncul dalam bahasa inggris yakni ‘THE’, maka pada saat mensubtitusi ‘ZPX’ dengan ‘THE’, maka dengan metode-metode selanjutnya yang cukup panjang, masih akan dapat ditemukan kunci awal yakni ‘LANGIT’. Dengan menggunakan metode dekripsi seperti pada bagian 2.3, maka cipherteks akan dapat dipecahkan.

3.2. Perbandingan dengan Vigenere Cipher

Setelah dilakukan kriptanalisis terhadap cipherteks sebelumnya, terdapat beberapa perbedaan yang dialami penulis dalam melakukan kriptanalisis, yakni dalam hal penebakan kunci. Pada vigenere cipher, kunci biasanya dibuat dalam bentuk kata yang mudah diingat, sehingga penebakan bisa lebih gampang dilakukan oleh kriptanalis.

Akan tetapi pada shif vigenere cipher ini, penebakan terasa lebih sulit karena kriptanalis tidak bisa menebak berapa panjang kunci sebenarnya. Kriptanalis hanya bisa mengetahui panjang kunci yang telah dimodifikasi oleh algoritma shift vigenere tersebut.

3.3. Saran dan Arah Pengembangan

Setelah melakukan kriptanalisis kepada cipherteks hasil enkripsi algoritma ini, maka diperlukan berbagai masukan untuk pengembangan metode ini lebih lanjut:

a. Mengembangkan lagi proses pembangkitan kunci dengan menambahkan metode lain sehingga pola pembentukan kunci tidak terlalu kelihatan seperti algoritma shift vigenere yang dirancang penulis sekarang
b. Menambahkan metode tertentu dalam proses enkripsi sehingga panjang plainteks tidak sama dengan panjang cipherteks sehingga akan lebih memperkuat algoritma ini

4. KESIMPULAN

Dari hasil pembahasan sebelumnya maka penulis menarik beberapa kesimpulan sebagai berikut:

a. Algoritma shift vigenere merupakan pengembangan metode vigenere cipher dengan modifikasi pada kunci
b. Algoritma ini masih bisa dipechkan dengan metode kasiski dan teknik analisis frekwensi
c. Algoritma shift cipher masih memiliki tingkat keamanan yang rendah, sehingga perlu dikembangkan lebih lanjut lagi.

Dalam penyusunan makalah ini, penulis mengucapkan terima kasih yang sebesar-besarnya kepada Pak Rinaldi Munir yang telah memberikan kesempatan bagi penulis untuk menyelesaikan makalah ini, walaupun sempat terjadi penundaan penyelesaian makalah yang dialami penulis karena faktor tertentu.

DAFTAR REFERENSI

[1] Munir, Rinaldi. (2006). Diktat Kuliah IF5054 Kriptografi. Program Studi Teknik Informatika Sekolah Teknik Elektro dan Informatika , Institut Teknologi Bandung.

[2] Microsoft Visual Studio 2005 Documentation