Modifikasi Nihilist Chiper

Fata Mukhlish[1]

Sekolah Teknik Elektro dan Informatika
Program Studi Teknik Informatika
Institut Teknologi Bandung
Jl. Ganesha 10 Bandung 40132

E-mail : if14084@students.if.itb.ac.id[1]

Abstraksi

Seiring meningkatnya perkembangan dunia teknologi, sistem pengaman yang canggih terhadap suatu data semakin dibutuhkan. Hal ini juga didorong oleh semakin maraknya kejahatan di dunia cyber. Salah satu sektor yang rawan mengundang kejahatan adalah pengiriman data. Oleh karena itu, pengguna teknologi semakin beramai-ramai mengembangkan suatu sistem pengamanan terhadap data yang biasa disebut kriptografi. Salah satu metode kriptografi yang dikenal adalah algoritma kriptografi klasik dimana metode ini sudah dikenal sejak lama. Algoritma kriptografi klasik terdiri dari 2 jenis, yang salah satu jenisnya adalah *Chiper Substitusi*, dimana yang akan dibahas dalam makalah ini adalah tipe *polyalphabetic chiper*, yaitu Nihilist cipher. Dalam makalah ini, penulis akan mencoba mengembangkan algoritma Nihilist cipher dengan cara memodifikasi cara kerja enkripsinya.

Kata kunci *:*
kriptografi,nihilist,polyalphabetic,chiper substitusi

1. Pendahuluan

Dalam dunia kriptografi, salah satu faktor suatu algoritma dikatakan aman adalah jika untuk memecahkannya dibutuhkan waktu dan biaya yang relatif besar. Salah satu jenis algoritma yang banyak digunakan yang memenuhi faktor ini adalah *polyalphabetic substitution chipper*, yang merupakan salah satu jenis dari Cipher Subtitusi. Cipher substitusi adalah tipe enkripsi pesan yang intinya adalah mengubah isi dari pesan dengan teks lain. Cipher subsitusi dapat dikelompokkan berdasarkan jumlah karakter hasil enkripsi relatif dibandingkan plainteksnya. Klasifikasi tersebut antara lain :

1. **Cipher abjad-tunggal** (*monoalphabetic cipher*)
   - Satu karakter dalam plainteks diganti dengan satu karakter yang bersesuaian sehingga fungsi enkripsi-dekripsinya satu ke satu.
   - Jika plainteks terdiri dari huruf abjad, maka jumlah kemungkinan susunan huruf-huruf cipherteks yang dapat dibuat adalah sebanyak 26!.
   - Sedangkan jika terdiri dari karakter ASCII maka kemungkinannya menjadi 256!.
   - Caesar cipher adalah kasus khusus dari cipher abjad tungal di mana susunan huruf cipherteks diperoleh dengan menggeser huruf-huruf alfabet sejauh tiga karakter.
   - Jumlah kunci di dalam cipher abjad-tunggal sama dengan jumlah cara menyususn ke-26 huruf abjad tersebut, yaitu sebanyak 26! yang juga menyatakan jumlah kunci untuk menyusun huruf-huruf alfabet ke dalam tabel substitusi.
   - Contoh tabel substitusi :
     P_i : A B C D E F G H I J K L M
     C_i : D I Q M T B Z S Y K V O F

     P_i : N O P Q R S T U V W X Y Z
     C_i : E R J A U W P X H L C N G
   - Sehingga untuk plainteks "**KRIPTOGRAFI**" dienkripsi menjadi "**VUYJPRZUDBY**".
2. **Cipher substitusi homofonik** (*homophonic substitution cipher*)
   - Seperti cipher abjad tunggal tetapi setiap karakter plainteks dapat dipetakan menjadi salah satu karakter cipherteks yang mungkin.
   - Fungsi enkripsi-dekripsi memetakan satu ke banyak.
   - Cipher substitusi homofonik pertama kali ditemukan pada tahun 1401 oleh wanita bangsawan Mantua.
   - Cipher substitusi homofonik lebih sulit dipecahkan daripada cipher abjad tunggal. Namun, dengan *known-plaintext attack* dapat dipecahkan sedangkan dengan *ciphertext-only attack* lebih sulit.
3. **Cipher abjad majemuk** (*Polyabathetic substitution cipher*)

- Merupakan cipher substitusi ganda yang melibatkan kunci berbeda.
- Cipher abjad majemuk dibuat dari sejumlah cipher abjad tunggal, masing-masing dengan kunci yang berbeda.
- Beberapa algoritma cipher jenis ini menggunakan Polibyus Square, salah satunya Nihilist cipher.

Dalam makalah ini, penulis akan membahas salah satu jenis *Polyabathetic substitution cipher* , yaitu Nihilist Cipher. Algoritma ini memiliki keunggulan karena setiap huruf yang sama belum tentu dienkripsi dengan kunci yang sama, sehingga dapat menghasilkan chiperteks yang berbeda. Hasil ini sudah tentu menyulitkan kriptanalis untuk memecahkan chiperteks hasil enkripsi metode ini.

2. Nihilist Cipher

2.1. Sejarah Singkat

Nihilist cipher pertama kali dikembangkan oleh para Russian Nihilist, yaitu orang-orang Rusia yang mendukung cara kekerasan untuk mencapai perubahan politik yang diinginkan, dalam hal ini menggulingkan kekuasaan Tsar Alexander II di Rusia. Mereka memanfaatkan algoritma Nihilist untuk berkomunikasi dan mengorganisasikan para teroris untuk melawan para pendukung Tsar pada tahun 1880-an [2].
Selain itu, algoritma ini juga banyak digunakan oleh First Chief Directorate, sebuah divisi dari KGB (badan intelejen Rusia) untuk berkomunikasi para calon mata-mata mereka. Serta digunakan pula untuk berkomunikasi dengan para sekutu mereka [2].

2.2. Konsep Dasar

Algoritma ini menggunakan Polybius Square (lihat Tabel 1), yaitu sebuah kotak, biasanya bujursangkar 5x5, mengacu pada huruf Latin, dengan menghilangkan huruf J dari abjad. Setiap elemen bujursangkar berisi huruf yang berbeda satu sama lain, yang dapat direpresentasikan dengan 2 digit koordinat yang berkaitan dengan elemen yang bersangkutan. Penempatan setiap huruf pun dapat diacak, tidak perlu berurutan.

	1	**2**	**3**	**4**	**5**
1	A	B	C	D	E
2	F	G	H	I	K
3	L	M	N	O	P
4	Q	R	S	T	U
5	V	W	X	Y	Z

Tabel 1: Polybius Square

Sehingga, sebagai contoh, kata AKU dapat direpresentasikan sebagai himpunan koordinat (11 25 45). Berikut dijelaskan mengenai langkah-langkah dalam melakukan enkripsi pada algoritma Nihilist dengan asumsi bahwa huruf yang digunakan adalah huruf Latin.

Langkah I:
Persiapkan 2 kata kunci, dengan syarat:
- Kata Kunci I : ≤ 25 huruf
- Kata Kunci II : ≤ plainteks

- **Langkah II**:
Misalkan, kata kunci I adalah KUNCI, masukkan kata ini ke dalam baris I Polybius Square, kemudian diikuti dengan huruf lainnya yang belum ada dalam kata kunci

	1	**2**	**3**	**4**	**5**
1	K	U	N	C	I
2	A	B	D	E	F
3	G	H	L	M	O
4	P	Q	R	S	T
5	V	W	X	Y	Z

Tabel 2: Tabel kunci

- **Langkah III**:
Lalu, misalkan kata kunci II adalah KRIPTO, kemudian berdasarkan tabel kunci diatas maka, koordinat yang berkoresponden dengan kata kunci II adalah (11 43 15 41 45 35).

K	R	I	P	T	O
11	43	15	41	45	35

Gambar 1: Koordinat Kata Kunci II

- **Langkah IV:**
Misalkan plainteksnya adalah MATA KULIAH, kemudian lakukan langkah III pada plainteks, sehingga:

M	A	T	A	K	U	L	I	A	H
34	21	45	21	11	12	33	15	21	32

Gambar 2: Koordinat plainteks

- **Langkah V:**
Lakukan operasi pertambahan antara koordinat plainteks dengan kata kunci II, sehingga akan didapat cipherteks:

kt	11	43	15	41	45	35	11	43	15	41
pt	34	21	45	21	11	12	33	15	21	32
ct	45	64	60	62	56	47	44	58	36	73

Gambar 3:Hasil Chiperteks dengan Nihilist

2.3. Kelebihan dan Kekurangan

- **Kelebihan:**

a) Mencegah frekuensi koordinat yang sama pada cipherteks, karena setiap huruf pada plainteks besar kemungkinan akan dienkripsi dengan huruf kata kunci yang berbeda.

b) Untuk memecahkannya membutuhkan waktu dan biaya yang tidak sedikit.

c) Penggunaan 2 kata kunci juga menghambat kriptanalis untuk memecahkan cipherteks.

- **Kekurangan:**

a) jika hasil digit pada chiperteks digit akhirnya 0, maka kedua huruf pada plainteks dan kunci berasal dari kolom ke-5.

b) jika digit chiperteks > 100, berarti digit plainteks dan kunci adalah 55

3. Modifikasi Nihilist Cipher

3.1. Konsep Umum

a) Proses Enkripsi

Secara umum, modifikasi ini dilalakukan dengan cara memodifikasi cara enkripsi dengan dekripsinya. Seperti dijelaskan sebelumnya, Nihilist Cipher melakukan enkripsinya dengan cara menjumlahkan koordinat plainteks dengan koordinat kata kunci yang diberikan. Sementara pada modifikasi yang dilakukan penulis, enkripsi dilakukan dengan cara mengurangkan koordinat plainteks dengan koordinat kata kunci yang diberikan, kemudian ditambahkan dengan suatu bilangan M. Sehingga secara matematis dapat dituliskan:

$$C_i = E(p_i) = [\ P_{ki} - K_{ki}\ ] + M \qquad [A]$$

$$M = [\ (K_1 + K_2) \bmod (|K_1 - K_2|)\ ] \qquad [B]$$

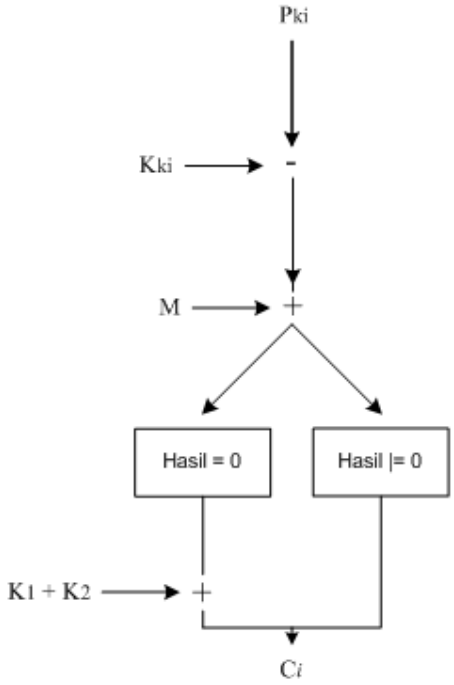
C_i = Cipherteks huruf ke-i
$E(p_i)$ = Enkripsi plainteks huruf ke-i
P_{ki} = Koordinat huruf plainteks ke-i
K_{ki} = Koordinat hurur kata kunci ke-i
K_1 = Jumlah Kata Kunci 1
K_2 = Jumlah Kata Kunci 2
M = Jika hasil sama dengan 0,ditambahkan **K1 + K2**

Perlu dijelaskan disini, bahwa bilangan M adalah suatu bilangan yang merupakan hasil penambahan jumlah kata kunci I dengan kata kunci II modulo pengurangan jumlah kata kunci I dengan kata kunci II secara absolut. Namun, apabila hasil modulo sama dengan 0, maka hasil ditambahkan dengan penambahan jumlah kata kunci I dengan kata kunci II. Hal ini sudah tentu dilakukan untuk mempersulit kriptanalis menyelesaikan masalah.

Sebagai contoh disini, misalkan kata kunci I adalah UJIAN, sedangkan kata kunci II adalah KRIPTOGRAFI. Maka, dengan menggunakan persamaan [B], secara matematis dapat dituliskan:
$K_1 = 5$
$K_2 = 11$
$M = (5 + 11) \bmod (|5 - 11|) = 4$

Maka, bilangan M sama dengan 4.

Berikut digambarkan diagram aliran data pada proses enkripsi:



Gambar 4: Skema Enkripsi

Untuk contoh permasalahan akan mengacu pada contoh langkah enkripsi Nihilist Cipher (lihat Hal. 2). Setelah kita mendapatkan koordinat plainteks dan koordinat kata kunci II:

- **Langkah V:**

kt	11	43	15	41	45	35	11	43	15	41
pt	34	21	45	21	11	12	33	15	21	32

Maka, kita menentukan bilangan M. karena kata kunci I adalah KUNCI, sedangkan kata kunci II adalah KRIPTO. Maka, secara matematis dapat dituliskan:
$K_1 = 5$
$K_2 = 6$
$M = (5 + 6) \bmod (|5 - 6|) = 0$

Disini terlihat hasil modulo 0, maka bilangan M-nya adalah $K_1 + K_2 = 11$.

- **Langkah VI:**

Lakukan proses enkripsi sesuai persamaan [A], sehingga akan dihasilkan cipherteks:

kt	11	43	15	41	45	35	11	43	15	41
pt	34	21	45	21	11	12	33	15	21	32

Ct	34	**11**	41	**09**	**23**	**12**	33	**17**	17	02

Gambar 5: Hasil Enkripsi dengan Modifikasi Nihilist

Sehingga akan kita dapatkan hasil cipherteks = 34**11**41**09231**233**17**1702

Perlu diperhatikan, untuk membedakan koodinat negatif dengan positif, penulis memberi tanda angka tebal (bold) untuk angka negatif. Sedangkan jika hasil cipherteks adalah angka satuan, maka ditambahkan angka 0 didepannya.

b) Proses Dekripsi

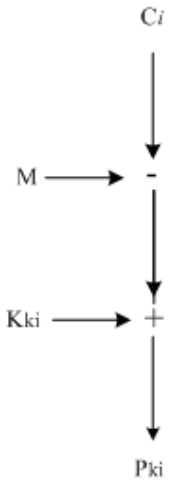
Untuk proses dekripsi, dilakukan proses secara matematis sebagai berikut:

$$p_i = D(C_i) = (C_i - M) + K_{ki} \qquad [C]$$

D(C*i*) = Dekripsi cipherteks huruf ke-*i*

Untuk menggunakan rumus [C] di atas, penerima pesan sudah tentu harus mendapatkan Kata Kunci I dan Kata Kunci II untuk membangun Polybius Square dan menentukan bilangan M.

Berikut digambarkan skema aliran data pada proses dekripsi:



Gambar 6: Skema Dekripsi

Sebagai contoh permasalahan, penulis akan mencoba untuk mendekripsi hasil enkripsi pada halaman 3, yaitu: 34**11**41**09231**233**17**1702

- **Langkah I:**

Ikuti langkah II dan langkah III dalam mengenkripsi dengan Nihilist Cipher, yaitu membangun Polybius Square dan mmbuat koordinat Kata Kunci II (Lihat Hal.2).

- **Langkah II:**

Ikuti langkah V dalam mengenkripsi dengan enkripsi modifikasi Nihilist Cipher (Lihat. Hal 3., yaitu mencari bilangan M dengan bantuan Kata Kunci I dan Kata Kunci II, sehingga didapatkan bilangan M = 11.

- **Langkah III:**

Pisahkan hasil cipherteks setiap 2 digit karakter untuk memudahkan melakukan proses dekripsi, sehingga:

Ct	34	**11**	41	**09**	**23**	**12**	33	**17**	17	02

- **Langkah IV:**

Lakukan proses dekripsi dengan mengikuti persamaan [C], sehingga didapatkan:

Ct	34	**11**	41	**09**	**23**	**12**	33	**17**	17	02
kt	11	43	15	41	45	35	11	43	15	41
pt	34	21	45	21	11	12	33	15	21	32

Gambar 7: Hasil Dekripsi dengan Modfikasi Nihilist

- **Langkah V:**

Setelah mendapatkan himpunan koordinat plainteks, selanjutnya dengan menggunakan Polybius Square, kita dapat mnetukan huuf yang cocok dengan kordinat yang berkaitan, sehingga didapatkan plainteks: MATA KULIAH.

3.2. Kelebihan dan Kelemahan

- **Kelebihan**

Kelebihan modifikasi ini terletak pada penggunaan 2 kata kunci dan bilangan M, hal ini disebabkan:

a. Penggunaan bilangan M yang berasal data Kunci I dan Kunci II menyulitkan kriptanalis untuk mendekripsi cipherteks.
b. Penggunaan 2 kata kunci juga menyulitkan kriptanalis, sehingga misalkan saja kriptanalis mendapatkan bilangan M dan koordinat kata kunci II, tanpa mendapatkan kata kunci I, mereka tidak dapat mendapatkan huruf yang sesuai dengan Polybius Square.
c. Kekurangan yang terdapat dalam algoritma Nihilist sebelumnya juga dapat diminimalisir, karena menggunakan bilangan M ini.
d. Algoritma enkripsi cukup sederhana sehingga mempercepat proses enkripsi.

- **Kelemahan**

Di samping memiliki kelebihan seperti sudah dijelaskan di atas, algoritma ini memiliki sedikit kelemahan-kelemahan sebagai berikut:

a. Jika bilangan M yang dihasilkan sangat kecil, seperti 1 atau 2, untuk kasus hasil **P*ki*** – **K*ki*** = -44 atau -43, maka bila hasil cipherteks (untuk M = 1) -43 atau -42. Hasil seperti ini dapat dengan mudah bagi kriptanalis untuk menerka bahwa bilangan M = 1.

b. Penerkaan panjang Kata Kunci II dapat dilakukan dengan metode Kasiski. Hal ini sedikit mempermudah menebak bilangan M.

c. Metode enkripsi ini hanya dapat digunakan untuk plainteks dengan huruf alphabet saja da tidak menggunakan simbol lainnya.

3.3. Tingkat Keamanan

Berikut akan diuraikan tingkat keamanan yang dilakukan beberapa teknik serangan kriptanalis terhadap Modifikasi Nihlist:

3.3.1. Ciphertext-only attack

Algoritma ini sangat kuat terhadap serangan ini, karena penggunaan bilangan M, dan 2 Kata Kunci. Selain itu, karena algoritma ini termasuk jenis *Polyabathetic substitution cipher*, maka sangat sult bagi kriptanalis untuk mendekripsikan cipherteks.

3.3.2. Known-plaintext attack

Algoritma in cukup kuat menghadapi seragan jenis ini, karena tanpa kriptanalis mengetahui kata kunci, terutama kata kunci I, Polybiu Square yang digunakan tidak dapat diketahui. Hal ini menyebabkan kriptanalis membutuhkan usaha lebih unutk memecahkan enkripsinya.

3.3.3. Chosen-plaintext attack

Karena algoritma ini cukup bertahan dengan*Known-plaintext attack*, maka pada serangan jenis ini juga algoritma modifikasi Nihilist masih cukup diandalkan.

3.3.4. Exhaustive attack

Untuk serangan jenis ini, algoritma penulis menunjukkan kelemahan, bila kriptanalis mengetahui panjang kunci, apalagi bila Kata Kunci I yang digunakan pendek. Bila panjang Kunci I diketahui, misalkn n karakter, maka jumlah kunci yang harus dicoba dilakukan 26^n hingga 26^{25} percobaan.

4. Kesimpulan dan Saran

Berdasarkan analisis dan perhitungan yang telah diuraikan di atas, dapat diambil kesimpulan:

a. Algoritma modifikasi Nihilist lebih baik daripada algoritma Nihilist Cipher biasa.
b. Algorima ini lebih kompleks karena menggunakan bilangan M.
c. Algoritma ini cukup kuat pada beberapa serangan seperti *ciphertext-only attack,plaintext-only attack, chosen-plaintext attack*.
d. Lemah terhadap exhaustive attack bila panjang kata kunci pendek.
e. Saran bagi pengguna yang ingin menggunakan algoritma ini adalah usahakan untul memakai kunci yang panjang.
f. Saran bagi yang ingin mengembangkan algoritma ini, kompleksitas dipersulit lagi.

5. Daftar Referensi

[1] Munir, Rinaldi, *Kriptografi*, Institut Teknologi Bandung, 2006.

[2] Kahn, David, *The Codebreakers*. 1968, 1974 edition Redwood Burn Ltd. pp 344, 368**.**

[3] Bucknell University, *Topics in Computer Science Fundamentals of Computer Security,* http://www.eg.bucknell.edu/~cs379/CompSec/2006-spring/lectures/lecture4.pdf.

[4]http://www.animal.ahrgr.de/showAnimationDetails.php3?lang=en&anim=215

[5]http://www.und.nodak.edu/org/crypto/crypto/.chap08.html