Modifikasi Vigènere Cipher Dengan Pendekatan Mode Operasi Cipher Block Chaining

Eko Budhi Susanto[1]
Departemen Teknik Informatika
Institut Teknologi Bandung
Jl. Ganesha 10 Bandung 40132
E-mail : if14075@students.if.itb.ac.id[1]

Abstraksi

Vigènere Cipher, salah satu algoritma kriptografi klasik yang menerapkan metode cipher subtitusi abjad-majemuk, saat ini sudah tidak lagi digunakan setelah berhasil dipecahkan pada tahun 1863 menggunakan metode Kasiski. Makalah ini membahas perancangan sebuah algoritma yang merupakan modifikasi dari Vigènere Cipher. Algoritma yang dirancang akan diujikan dengan metode yang pernah digunakan oleh Kasiski untuk memecahkan Cipher ini satu setengah abad silam. Algoritma ini dirancang sedemikian rupa menggunakan pendekatan mode operasi Cipher Block Chaining (CBC) yang biasa digunakan untuk penyandian dalam mode bit. Algoritma ini kemudian akan kita namakan Chaining Text Vigènere Cipher.

kata kunci *: Kriptografi klasik, Algoritma, cipher subtitusi abjad-majemuk, Vigènere Cipher, Metode Kasiski, Cipher Block Chaining, Chaining Text Vigènere Cipher*

1. Pendahuluan

Sebelum adanya komputer, kriptografi dilakukan berbasis karakter dengan hanya menggunakan kertas dan pena. Algoritma kriptografi yang digunakan saat itu termasuk dalam sistem kriptografi kunci simetri dan digunakan jauh sebelum ditemukannya sistem kunci publik. Algoritma kriptografi yang berbasis karakter biasanya termasuk dalam salah satu dari cipher substitusi, cipher transposisi, atau super enkripsi yang merupakan gabungan dari cipher substitusi dan cipher transposisi.

Jika berbicara mengenai kriptografi klasik, pastilah banyak yang sudah mengetahui Vigènere Cipher, salah satu algoritma kunci simetri yang menerapkan metode cipher subtitusi abjad-majemuk. Algoritma ini pertama kali dikenalkan oleh Bastia Belaso dalam bukunya *La cifra del. Sig. Giovan Batista Belaso* (1553), lalu disempurnakan oleh diplomat Perancis Blaise de Vigènere pada tahun 1586. Pada abad ke-19 banyak orang mengira Vigènere adalah penemu cipher ini, sehingga sampai sekarang cipher ini dikenal luas sebagai Vigènere Cipher. Vigènere Cipher saat ini sudah tidak lagi digunakan setelah Freidrich Kasiski menemukan pemecahannya pada tahun 1863 menggunakan metode yang dinamakan metode Kasiski.

Jika dilihat dari aspek keamanan standar yang diberikan oleh kriptografi, Vigènere Cipher hanya menyediakan jaminan **kerahasiaan** (*confidentiality*) dari 4 jaminan yang seharusnya disediakan**. Jaminan integritas data** (*data integrity*), **otentikasi** (*authentication*), dan **Nirpenyangkalan** (*non-repuditation*) belum disediakan oleh cipher yang satu ini, mengingat Vigènere Cipher adalah sebuah algoritma klasik.

Untuk menambah aspek keamanan kriptografi pada Cipher ini, maka dilakukan sedikit modifikasi, yaitu dengan cara menerapkan salah satu metode yang ada pada salah satu algoritma kriptograri modern. Metode yang akan diterapkan pada Cipher ini adalah mode operasi **Cipher Block Chaining** (CBC) yang merupakan salah satu mode operasi yang digunakan oleh algoritma kriptografi modern Cipher blok.

Algoritma baru yang kelak dinamakan **Chaining Text Vigènere Cipher** ini memiliki keunggulan dibanding pendahulunya, algoritma Vigènere Cipher standar. Ada beberapa kelebihan dan kekurangannya yang akan dibahas pada makalah ini, tetapi yang utama, algoritma ini menyediakan 2 aspek keamanan dari 4 aspek yang seharusnya dimiliki oleh sebuah algoritma kriptografi, yaitu aspek kerahasiaan, dan integritas data.

2. Vigènere Cipher dan Analisis Metode Kasisiki

Algoritma enkripsi Vigènere Cipher pertama kali dikenalkan oleh **Bastia Belaso** dalam bukunya *La cifra del. Sig. Giovan Batista Belaso* (1553), lalu disempurnakan oleh diplomat Perancis **Blaise de Vigènere** pada tahun 1586. Pada abad ke-19 banyak orang mengira Vigènere adalah penemu cipher ini, sehingga sampai sekarang cipher ini dikenal luas sebagai Vigènere Cipher.

Vigènere Cipher adalah sebuah algoritma enkripsi yang menggunakan konsep **Caesar Cipher** dengan kunci yang berbeda-beda. Vigènere cipher adalah contoh sederhana dari algoritma enkripsi subtitusi abjad majemuk (*Polyalphabetic Subtitution Cipher*). Algoritma enkripsi ini terkenal karena selain mudah

dimengerti dan diimplementasikan, algoritma ini juga bagi kriptanalis pemula tampak seperti tidak dapat dipecahkan.

Vigènere Cipher mendapatkan reputasi sebagai algoritma enkripsi yang kuat. Seorang ilmuwan matematika, **Charles Lutwidge Dodgson**, menyebut Vigènere Cipher sebagai *unbreakable cipher* pada sebuah majalah anak-anak tahun 1868. Pada tahun 1917, ilmuwan amerika menggambarkan Vigènere Cipher dengan "*impossible of translation*". Reputasi ini lepas setelah Kasiski dengan tuntas memecahkan Vigènere Cipher pada abad ke 19.

Vigènere Cipher digunakan oleh tentara Konfiderasi pada Perang Sipil Amerika. Perang sipil terjadi setelah Vigènere Cipher berhasil dipecahkan.

Vigènere Cipher menggunakan Bujursangkar Vigènere untuk melakukan enkripsi. Setiap baris di dalam bujursangkar menyatakan huruf-huruf cipherteks yang diperoleh dengan Caesar Cipher. Bentuk bujursangkar Vigènere adalah seperti terlihat pada **garmbar 1**. Deretan alpabet kuning mendatar merepresentasikan plainteks, sedangkan deretan alpabet kuning menurun merepresentasikan kunci.

Maka dapat dilihat pada tabel Vigènere bahwa untuk huruf pertama, T yang berkorespondensi dengan huruf S pada kunci akan menghasilkan huruf L.

Hasil enkripsi seluruhnya adalah:

```
Plainteks     : THIS PLAINTEXT
Kunci         : sony sonysonys
Cipherteks    : LVVQ HZNGFHRVL
```

Pada dasarnya, setiap enkripsi huruf adalah Caesar cipher dengan kunci yang berbeda-beda.
c('T') = ('T' + 's') mod 26 = L
c('H') = ('H' + 'o') mod 26 = V, dst.

Kelebihan Algoritma Vigenere

Pada algoritma Vigènere Cipher, huruf yang sama pada plainteks tidak selalu menjadi huruf yang sama pula dalam cipherteks. Pada contoh diatas huruf plainteks T dapat dienkripsi menjadi L atau H, dan huruf cipherteks V dapat merepresentasikan H, I, dan X dalam plainteks.

Vigènere Cipher dapat mencegah frekuensi huruf di dalam cipherteks yang mempunyai pola tertentu yang sama seperti pada cipher abjad tunggal. Jika

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
A	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
B	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
C	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
D	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
E	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
F	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
G	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
H	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
I	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
J	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
K	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
L	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
M	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
N	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
O	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
P	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
Q	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
R	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
S	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
T	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
U	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
V	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
W	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
X	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
Y	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
Z	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y

gambar 1 : ***Bujursangkar Vigènere***

Jika panjang kunci lebih pendek daripada panjang plainteks, maka kunci diulang secara periodik. Bila panjang kunci adalah m, maka periodenya dikatakan m.

Contoh:

```
Kunci       : sony
Plainteks   : this plaintext
Kunci       : sony sonysonys
```

periode kunci diketahui dan tidak terlalu panjang, maka kunci dapat ditentukan dengan menulis program komputer untuk melakukan *exhaustive key search*.

Contoh, diberikan cipherteks sebagai berikut :

```
CSASTP KV SIQUT GQU CSASTPIUAQJB
```

dan diperoleh informasi bahwa panjang kunci adalah *p* huruf dan plainteks ditulis dalam Bahasa Inggris, maka *running* program dengan mencoba semua

kemungkinan kunci yang panjangnya *p* huruf, lalu periksa apakah hasil dekripsi dengan kunci tersebut menyatakan kata yang berarti. Cara ini membutuhkan usaha percobaan sebanyak 26*p* kali.

Metode Kasiski

Metode Kasiski membantu menemukan panjang kunci Vigènere Cipher. Metode Kasiski memanfaatkan keutungan bahasa inggris yang tidak hanya mengandung perulangan huruf, tetapi juga perulangan pasangan huruf atau tripel huruf seperti TH, THE, dsb. Perulangan huruf ini memungkinkan menghasilkan kriptogram yang berulang.

Contoh:
Plainteks:
`CRYPTO IS SHORT FOR CRYPTOGRAPHY`
Kunci :
`abcdab cd abcda bcd abcdabcdabcd`
Cipherteks:
`CSASTP KV SIQUT GQU CSASTPIUAQJB`

Pada contoh tersebut, subkata CRYPTO secara kebetulan dienkripsi menjadi kriptogram yang sama, yaitu CSATP. Hal ini dikarenakan jarak antara dua buah string yang berulang di dalam plainteks merupakan kelipatan dari panjang junci, maka string yang sama tersebut akan muncul menjadi kriptogram yang sama pula dalam cipherteks. Tujuan metode Kasiski adalah mencari dua atau lebih kriptogram berulang untuk menentukan panjang kunci.

Langkah-langkah metode Kasiski:

1. Temukan semua kriptogram yang berulang di dalam cipherteks (pesan yang panjang biasanya mengandung kriptogram yang berulang).
2. Hitung jarak antara kriptogram yang berulang
3. Hitung semua faktor (pembagi) dari jarak tersebut (faktor pembagi menyatakan panjang kunci yang mungkin).
4. Tentukan irisan dari himpunan faktor pembagi tersebut. Nilai yang muncul di dalam irisan menyatakan angka yang muncul pada semua faktor pembagi dari jarak-jarak tersebut . Nilai tersebut mungkin adalah panjang kunci. Hal ini karena *string* yang berulang dapat muncul bertindihan (*coincidence*)

Setelah panjang kunci diketahui, maka langkah berikutnya adalah menentukan kata kunci. Kata kunci dapat ditentukan dengan menggunakan metode *exhaustive key search*. Jika panjang kunci adalah *p*, maka jumlah kunci yang harus dicoba adalah 26^p. Namun akan lebih efesien bila menggunakan teknik analisi frekuensi.

Langkah-langkah untuk melakukan analisis frekuensi adalah:

1. Misalkan panjang kunci yang sudah berhasil dideduksi adalah *n.* Setiap huruf kelipatan ke-*n* pasti dienkripsi dengan huruf kunci yang sama. Kelompokkan setiap huruf ke-*n* bersama-sama sehingga kriptanalis memiliki *n* buah “pesan”, masing-masing dienkripsi dengan substitusi alfabet-tunggal (dalam hal ini *Caesar cipher*).
2. Tiap-tiap pesan dari hasil langkah 1 dapat dipecahkan dengan teknik analisis frekuensi.
3. Dari hasil langkah 3 kriptanalis dapat menyusun huruf-huruf kunci. Atau, kriptanalis dapat menerka kata yang membantu untuk memecahkan cipherteks

3. Konsep Mode Operasi *Chaining Block Cipher*

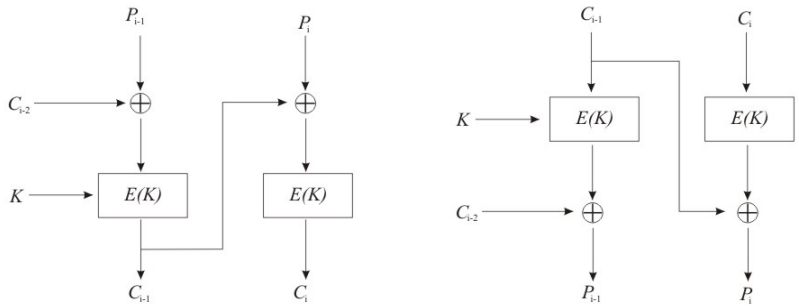
Chaining Block Cipher adalah salah satu mode operasi pada algoritma cipher blok. Algoritma cipher blok sendiri adalah salah satu algoritma kriptografi modern, yaitu algoritma yang beroperasi pada plaintext/ciphertext dalam bentuk blok bit, yang dalam hal ini rangkaian bit dibagi menjadi blok-blok bit yang panjangnya sudah ditentukan sebelumnya. Cipher blok ini dapat beroperasi pada 4 jenis mode. Yaitu *Electronic Code Book* (ECB), *Cipher Block Chaining* (CBC), *Cipher Feedback* (CFB), dan *Output Feedback* (OFB).

Konsep yang akan dibahas disini bukan tentang cipher blok, melainkan hanya tentang *Chaining Block Cipher*(CBC)-nya saja.

Mode ini menerapkan mekanisme umpan-balik (*feedback*) pada sebuah blok, yang dalam hal ini hasil enkripsi blok sebelumnya di-umpan-balik-kan ke dalam enkripsi blok *current*, yaitu blok yang sedang dioperasikan saat ini.

Caranya, blok plainteks yang *current* di-XOR-kan terlebih dahulu dengan blok cipherteks hasil enkripsi sebelumnya. Selanjutnya hasil peng-XOR-an masuk kedalam fungsi enkripsi. Dengan mode CBC, setiap blok cipherteks bergantung tidak hanya pada blok plainteksnya, tetapi juga pada seluruh blok plainteks sebelumnya.

Dekripsi dilakukan dengan memasukkan blok cipherteks *current* pada fungsi dekripsi, kemudian meng-XOR-kan hasilnya dengan blok cipherteks sebelumnya. Dalam hal ini, blok cipherteks sebelumnyaberfungsi sebagai umpan-maju (*feedforward*) pada akhir proses deksipsi. Skema mode CBC dapat dilihat pada **gambar 2** dibawah ini.



(a) enkripsi *(b) dekripsi*

gambar 2 : ***Skema Enkripsi dan Dekripsi dengan Mode CBC***

Dalam hal ini, C_0 = IV (*Initial Vector*). IV dapat diberikan oleh pengguna atau dibangkitkan secara acak oleh program. Jadi, untuk menghasilkan blok cipherteks pertama (C_1), IV digunakan untuk menggantikan blok cipherteks sebelumnya, C_0. Sebaliknya, pada dekripsi, blok plainteks diperoleh dengan meng-XOR-kan IV dengan hasil dekripsi terhadap blok cipherteks pertama.

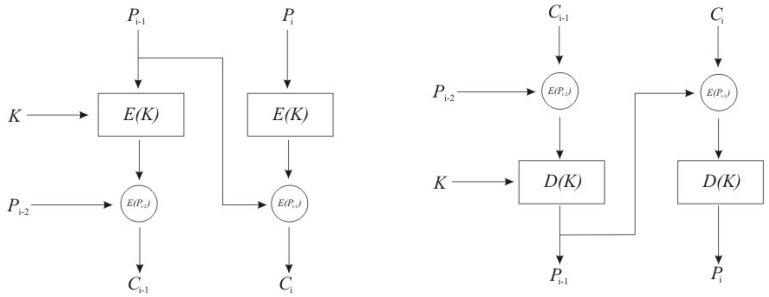
Keuntungan Mode CBC

Karena blok-blok plainteks yang sama tidak menghasilkan blok-blok cipherteks yang sama, maka proses kripanalisis menjadi lebih sulit. Inilah alasan utama penggunaan mode CBC digunakan.

4. Perancangan *Chaining Text Vigènere Cipher*

Secara umum, teknik enkripsi dan dekripsi pada *Chaining Text Vigènere Cipher* (CTVC) seperti pada Vigènere Cipher standar. Penerapan kunci, dan cara pemakaian kuncinya pun serupa. Modifikasi yang dilakukan untuk mengubah Vigènere Cipher standar menjadi CTVC adalah dengan menambahkan mode operasi yang mirip dengan CBC, tetapi dilakukan pada plainteks alpabet, dan tidak di-XOR-kan, melainkan di enkripsi seperti melakukan enkripsi pada Vigènere Cipher atau Caesar Cipher.

Teknik operasi yang digunakanpun dimodifikasi,



(a) enkripsi *(b) dekripsi*

gambar 3 : ***Skema enkripsi dan dekripsi dengan mode Reverse CBC***

Kelemahan Mode CBC

Karena blok cipherteks yang dihasilkan selama proses enkripsi bergantung pada blok-blok cipherteks sebelumnya, maka kesalahan satu bit pada sebuah blok plainteks akan merambat pada blok cipherteks yang berkoresponden dan semua blok cipherteks berikutnya. Hal ini berkebalikan dengan proses dekripsi, kesalahan satu bit pada blok cipherteks hanya mempengaruhi blok plainteks yang berkoresponden dan satu bit pada blok plainteks berikutnya (pada posisi bit yang berkoresponden pula).

menjadi seperti **gambar 3**. Yaitu Reverse CBC. Yaitu dengan mengenkripsi plainteks dengan kunci, lalu mengenkripsinya lagi dengan plainteks sebelumnya (yang berarti pada langkah awal menggunakan IV). Sedangkan proses dekripsinya adalah dengan mendekripsi cipherteks menggunakan plainteks sebelumnya sebagai kunci (atau IV pada langkah pertama), lalu mendekripsinya lagi menggunakan kunci yang sebenarnya.

Algoritma Enkripsi

Secara umum algoritma enkripsi CTVC adalah sebagai berikut:

1. Menentukan kunci dan IV. IV bisa diambil dari huruf pertama kunci, huruf terakhir kunci, atau

sebuah huruf lain yang dipilih secara acak. IV juga bisa berupa sebuah kata, atau kalimat. Jadi pergeseran C sejauh panjang IV (misalnya jika panjang IV = 3, maka C_i didapat dari plainteks P_{i-3}). Misalkan panjang IV = u.
2. Plainteks P_i dienkripsi menggunakan kunci.
3. Hasilnya dienkripsi lagi menggunakan A_i yang didapat dari P_{i-u} (pada plainteks awal menggunakan IV) sebagai kuncinya.
4. Kedua proses enkripsi dilakukan dengan menggunakan bujursangkar vigènere.

Contoh operasi:

Plainteks	: `contoh kasus`
Kunci	: `abcabc abcab`
(*hasil#1*)	`cpptpj kbuut`
A (IV = N)	: `nconto hkasu`
Cipherteks	: `prdgix rlumn`

Keterangan :
Hasil#1 adalah hasil yang didapat jika melakukan proses enkripsi dengan Vigènere Cipher standar.
Pada contoh ini, huruf c pada A_2 didapat dari huruf c pada plainteks P_1, begitu pula huruf o pada A_3 yang didapat dari huruf o pada plainteks P_2, demikian seterusnya.

Jika dibandingkan, cipherteks pada CTVC menghasilkan sususan huruf-huruf yang lebih acak dan sulit dipecahkan dibandingkan Vigènere Cipher standar.

Algoritma Dekripsi
Algoritma dekripsi untuk CTVC adalah sebagai berikut:
1. Cipherteks C_i didekripsi terlebih dahulu menggunakan A_i yang didapat dari plainteks P_{i-u}, hasil dekripsi sebelumnya (pada cipherteks awal digunakan IV) sebagai kuncinya. U dalah panjang IV.
2. Hasil dekripsi tadi didekripsikan lagi menggunakan kunci sebenarnya.
3. Kedua proses dekripsi dilakukan menggunakan bujursangkar vigènere.

Contoh operasi:

Cipherteks	: `prdgix rlumn`
A (IV = N)	: `nconto hkasu`
(*Hasil#1*)	: `cpptpj kbuut`
Kunci	: `abcabc abcab`
Plainteks	: `contoh kasus`

Keterangan :
Hasil#1 adalah hasil yang didapat jika melakukan proses dekripsi cipherteks dengan A. *Hasil#1* ini juga merupakan cipherteks hasil enkripsi Vigènere Cipher standar.

Salah satu keunggulan CTVC ini adalah, kesatuan datanya, jadi jika ada serangan dari pihak lawan, yaitu dengan cara mengubah cipherteks, menghilangkan beberapa cipherteks, atau menambah isi cipherteks, pihak penerima tidak akan mendapatkan plainteks yang dapat dimengerti, dengan kata lain plainteks dikatakan hilang. Dengan demikian, CTVC sudah memenuhi aspek keamanan kriptografi yang kedua, yaitu menjamin integritas data. Data yang dikirim akan diterima dalam keadaan utuh seperti aslinya, atau tidak dapat terbaca sama sekali.

Serangan Terhadap Cipherteks
Contoh serangan dengan cara pengubahan salah satu bagian cipherteks, misal:
Cipherteks asli : `prdgix rlumn`
Cipherteks yang telah mendapat serangan :
`Prdgis rlumn`
(Huruf x diubah menjadi s)

Cipherteks	: `prdgis rlumn`
A (IV = N)	: `nconto cpvxp`
(*Hasil#1*)	: `cpptpe pwzpy`
Kunci	: `abcabc abcab`
Plainteks	: `contoc pvxpx`

Contoh kedua adalah dengan cara penambahan cipherteks. Misalkan:
bagian cipherteks, misal:
Cipherteks asli : `prdgix rlumn`
Cipherteks yang telah mendapat serangan :
`Prdgix hrlumn`
(Penambahan huruf h diantara huruf x dan r)

Cipherteks	: `prdgix hrlumn`
A (IV = N)	: `nconto hakzvq`
(*Hasil#1*)	: `cpptpj arbvrx`
Kunci	: `abcabc abcabc`
Plainteks	: `contoh akzvqv`

Contoh ketiga, serangan dengan cara pengurangan satu atau lebih cipherteks, misalnya:
Cipherteks asli : `prdgix rlumn`
Cipherteks yang telah mendapat serangan :
`Prdgi rlumn`
(Huruf x dihilangkan)

Cipherteks	: `prdgi rlumn`
A (IV = N)	: `ncont obkjb`
(*Hasil#1*)	: `cpptp dkkdo`
Kunci	: `abcab cabca`
Plainteks	: `conto bkjbo`

Dari tiga contoh diatas, didapat, cipherteks yang telah ternoda oleh serangan tidak dapat menghasilkan plainteks yang mempunyai arti. Maka integritas data yang dikirim akan tetap terjamin.

Metode Kasiski
Metode kasiski digunakan untuk menemukan panjang kunci pada Vigènere Cipher dengan cara mencari kriptogram yang sama pada cipherteks.

Dengan menggunakan CTVC, kemungkinan adanya kriptogram berulang dapat diperkecil. Dari contoh dibawah ini kita bisa melihat perbedaan antara Vigènere Cipher standar dengan CTVC dalam mengenkripsi plainteks yang memiliki pasangan huruf berulang yang jaraknya merupakan kelipatan panjang kunci.

Contoh:
Plainteks:
`CRYPTO IS SHORT FOR CRYPTOGRAPHY`
Kunci :
`abcdab cd abcda bcd abcdabcdabcd`
Cipherteks menggunakan Vigènere Cipher:
`CSASTP KV SIQUT GQU CSASTPIUAQJB`
Cipherteks menggunakan CTVC (IV = a):
`CURQII YD KAXIK ZVI TURQIIWARQYI`

Jika IV diperpanjang, menjadi beberapa huruf, atau bahkan sebuah kata, maka kemungkinan ini bisa diminimalisir lagi, bahkan dihilangkan, yaitu dengan menggunakan IV yang ukurannya lebih panjang dari panjang maksimum kriptogram berulang yang ada dalam plainteks.

Contoh, menggunakan plainteks dan kunci seperti contoh sebelumnya, namun dengan IV = "fedcba". Panjang IV sama dengan panjang kriptogram berulang pada plainteks.
Plainteks:
`CRYPTO IS SHORT FOR CRYPTOGRAPHY`
Kunci:
`abcdab cd abcda bcd abcdabcdabcd`
(*hasil#1*)
`CSASTP KV SIQUT GQU CSASTPIUAQJB`
A (IV = fedcba):
`fedcba cr yptoi ssh ortforcrypto`
Cipherteks:
`HWDUUP MM QXJIB YIB MJTXHGKLYFCP`

Dapat dilihat hasilnya, cipherteks sekarang tidak menghasilkan kriptogram berulang, sehingga kripanalis akan susah menemukan korelasi huruf plainteks dengan cipherteks, dan panjang kunci tidak ditemukan walaupun dicari dengan metode kasiski, karena kriptogram berulang tidak ditemukan.

Kelebihan
Chaining Text Vigenere Cipher (CTVC) memiliki beberapa keunggulan jika dibandingkan dengan Vigènere Cipher standar. Dari segi keamanan, CTVC menawarkan penyandian yang lebih aman, dengan pengacakan yang lebih rumit, menggunakan IV yang panjang, seperti terlihat pada contoh diatas. Dari segi lainnya, CTVC memiliki keunggulan dibanding Vigènere Cipher standar, yaitu mendukung integritas data, sehingga aman dari serangan oleh pihak lawan.

Kekurangan
Walaupun begitu, CTVC tetap memiliki kekurangan. Diantaranya, rumitnya teknik enkripsi dan dekripsi, karena untuk sebuah plainteks, enkripsi dilakukan dua kali, pertama menggunakan kuncinya, kedua menggunakan IV-nya, atau A-nya (Dalam hal ini A adalah plainteks sebelumnya). Begitu pula dengan teknik dekripsi.

Untuk pengiriman data yang harus sampai walaupun terjadi sedikit cacat, kelebihan CTVC malah menjadi salah satu kekurangannya. Cipherteks yang ternoda karena serangan tidak dapat mengembalikan plainteks utuh, atau plainteks yang sedikit rusak atau ternoda seperti halnya pada Vigènere Cipher. Pada CTVC, bagian setelah Cipherteks ternoda (diubah, dihapus, maupun bagian setelah cipherteks palsu yang ditambahkan) tidak dapat didekripsi dengan benar.

5. Kesimpulan

Algoritma Vigènere biasa merupakan algoritma yang sangat kuat sebelum dipecahkan oleh Kasiski. Oleh karena itu, dengan menemukan sebuah algoritma berdasarkan pada Vigènere yang diolah sedemikian rupa hingga menyulitkan menemukan panjang kunci dengan menggunakan metode Kasiski akan membuat algoritma tersebut menjadi sebuah algoritma enkripsi yang kuat.

Algoritma *Chaining Text Vigènere Cipher* merupakan modifikasi dari Vigènere Cipher. Modifikasi yang dilakukan adalah untuk menyembunyikan panjang kunci, dengan penambahan IV, yang panjangnya relatif sesuai keinginan pemakainya. Selain itu, penambahan aspek keamanan juga menjadi hal yang dibahas dalam modifikasi ini. Dengan modifikasi konsep CBC yang diterapkan pada Vigènere ini, membuat integritas data cipherteks terjamin.

6. Referensi

- [MUN06] Munir, Rinaldi, *Kriptografi*, Institut Teknologi Bandung, 2006.
- [BIS03] Bishop, David, *Introduction to Cryptography with Java Applets*, Jones and Bartet, 2003.
- [WIK07] http://en.wikipedia.org/, diakses selama oktober 2007.