Analisis Serangan *Dictionary attack* pada Cipherteks Berbasis Substitusi Monoalfabetik

Haryus Aminul Akbar 13507016
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
If17016@if.itb.ac.id

Abstraksi— Pada kriptografi dengan metode-metode klasik (substitusi monoalfabetik, polialfabetik), teknik yang lazim digunakan untuk kriptanalisis cipherteks adalah teknik analisis frekuensi. Khusus pada substitusi monoalfabetik, terdapat suatu pendekatan untuk kriptanalisis yaitu dengan menggunakan serangan Dictionary attack. Pendekatan ini dimungkinkan karena pada substitusi monoalfabetik, suatu kata pada cipherteks yang bersesuaian dengan plainteksnya, masih memiliki pola kombinasi simbol yang sama antara plainteks dan cipherteks. Dengan melihat kesesuaian kombinasi simbol (dalam hal ini alfabet), kita dapat mencari pada kamus bahasa kata-kata apa saja yang bersesuaian dengan pola kombinasi huruf tersebut. Pada makalah ini akan dikaji metode serangan Dictionary attack pada cipherteks berbasis substitusi monoalfabetik, bagaimana optimalisasi penggunaannya dan kelebihan serta kekurangannya.

***Kata Kunci*— cipherteks, *dictionary attack*, kriptanalisis, kriptografi, substitusi monoalfabetik.**

I. PENDAHULUAN

Kriptografi sebagai seni untuk menyembunyikan makna dari pesan sudah digunakan sejak jaman dahulu, dan mungkin berusia sama tuanya dengan penemuan huruf sebagai alat untuk menyampaikan pesan dan pikiran.

Dengan adanya kepentingan pribadi dalam proses penyampaian pesan kepada seseorang, tidak semua pesan ingin dapat dibaca oleh semua orang. Kepentingan ini melahirkan ilmu kriptografi, yang bertujuan menyembunyikan suatu pesan agar tidak dapat dibaca semua orang yang tidak berkepentingan, baik dengan cara menyembunyikan makna dari pesan atau menyembunyikan keberadaan pesan itu sendiri. Selama kepentingan pribadi manusia tetap ada, selama itu pula lah seni kriptografi akan terus berkembang, dari jaman dulu sampai sekarang.

II. SUBSTITUSI MONOALFABETIK

A. Metode-metode Substitusi Monoalfabetik

Pada seni kriptografi klasik, diketahui beberapa metode penyandian yang pernah dikenal sejarah. Salah satu metode paling klasik yaitu sandi Atbash, yang digunakan bangsa Yahudi konon sejak sekitar 600 SM. Sandi Atbash mengganti alfabet Hebrew dengan korespondensi kebalikannya :

אבגדהוזחטיכלמנסעפצקרשת
תשרקצפעסנמלכיטחזוהדגבא

Jika diterapkan pada alfabet latin maka akan berupa:

abcdefghijklmnopqrstuvwxyz
zyxwvutsrqponmlkjihgfedcba

Pada model penyandian ini, huruf ‘a’ pada plainteks (yaitu teks asal yang belum disandikan) akan diubah menjadi huruf ‘z’ pada cipherteks (teks yang sudah disandikan), huruf ‘b’ akan disandikan dengan huruf ‘y’, dan seterusnya.

Salah satu lagi sistem penyandian klasik yang dikenal sejarah datang dari Romawi Kuno, yaitu *Caesar Cipher*. Pada penyandian *Caesar*, huruf pada cipherteks berkorespondensi dengan huruf pada plainteks yang urutannya digeser sebanyak tiga huruf :

abcdefghijklmnopqrstuvwxyz
defghijklmnopqrstuvwxyzabc

Maka apabila pesan pada plainteks berbunyi ‘danger’ maka pada cipherteks nya pesan tersebut berbunyi ‘gdqjhu’.

Dari daratan Asia juga dikenal suatu metode penyandian klasik. Vatsayana, penulis buku *Kama Sutra* (yang diperkirakan ditulis antara abad 1-4 M), menyatakan di dalam *Kama Sutra* bahwa *mlecchita-vikalpa*, seni dalam tulisan rahasia merupakan seni ke-45 dari 64 *yoga* yang harus dikuasai oleh para istri[1]. Kriptografi yang dijelaskan Vatsayana adalah membuat daftar kelompok dua huruf dari semua alfabet, dan mengganti huruf dalam plainteks dengan pasangan hurufnya dari daftar yang telah dibuat. Misal daftar pasangan huruf yang telah dibuat adalah :

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

Maka jika huruf yang hendak disandikan adalah ‘s’

maka pada cipherteks akan diganti ‘n’ dan demikian pula sebaliknya. pesan yang berbunyi ‘brahmastra’ akan disandikan menjadi ‘kmifrinymi’.

Metode-metode penyandian klasik seperti yang disebutkan di atas, memiliki satu kesamaan penting dalam proses penyandiannya. Setiap huruf pada plainteks akan disandikan menjadi tepat satu huruf yang lainnya. Metode ini secara umum disebut sebagai “Substitusi Monoalfabetik”. Pada pengembangannya, substitusi tiap huruf tidak harus huruf lainnya, tapi pada prinsipnya tiap huruf pada plainteks akan diubah ke dalam satu bentuk huruf atau simbol lain.

B. Analisis Frekuensi pada Substitusi Monoalfabetik

Kelemahan pada kriptografi dengan metode substitusi monoalfabetik adalah distribusi frekuensi huruf pada cipherteks akan merefleksikan distribusi frekuensi alfabet pada umumnya (alfabet pokok)[2]. Analisis ini pertama kali dikemukakan oleh matematikawan Arab, Al-Kindi, pada abad ke-9 M, dalam bukunya *A Manuscript on Deciphering Cryptographic Messages*[3]. Jika diketahui dalam suatu cipherteks huruf yang kemunculannya paling banyak adalah ‘x’, maka jika cipherteks tersebut ditulis dalam bahasa Inggris, kemungkinan huruf tersebut adalah ‘e’. Hal ini dikarenakan dalam bahasa Inggris, huruf yang paling sering digunakan adalah huruf ‘E’.

Kita ambil contoh sebuah sampel cipherteks dari situs http://crypto.lkdev.com. Untuk analisis frekuensi dari cipherteks dapat kita gunakan *tool* dari situs yang sama. Cipherteks yang kita selidiki menggunakan bahasa Inggris, yaitu :

MJ EIBO TEJCJ TSXTEC TI LJ CJBU JRVOJPT, TEWT WBB KJP WSJ ASJWTJO JQXWB, TEWT TEJY WSJ JPOIMJO LY TEJVS ASJWTIS MVTE AJSTWVP XPWBVJPWLBJ SVGETC, TEWT WKIPG TEJCJ WSJ BVUJ, BVLJSTY WPO TEJ NXSCXVT IU EWNNVPJCC. WPO TEWT TI CJAXSJ TEJCJ SVGETC, GIRJSPKJPTC WSJ VPCTVTXTJO WKIPG KJP, OJSVRVPG TEJVS ZXCT NIMJSC USIK TEJ AIPCJPT IU TEJ GIRJSPJO.

Dengan menghitung jumlah kemunculan tiap huruf dalam cipherteks, kita mendapatkan statistik sebagai berikut:

huruf	frekuensi
J	45
T	36
S	21
W	20
P	19
E	19
V	16
C	16
I	14
O	10
B	9
X	8
G	7
K	6
A	5
U	5
L	4
M	4
R	4
N	4
Y	3
Z	1
Q	1

Dalam bahasa Inggris, 10 huruf yang paling sering muncul adalah E-T-A-O-I-N-S-H-R-D-L-U. Dengan mencoba memasangkan huruf dengan frekuensi paling banyak pada cipherteks dengan 10 huruf yang paling sering muncul tersebut, kita bisa memulai mencoba memecahkan cipherteks tersebut. Tentu saja 10 huruf dengan frekuensi tertinggi tersebut mungkin tidak sesuai dengan daftar huruf yang paling sering muncul dalam bahasa Inggris di atas, akan tetapi paling tidak peluang 10 huruf tersebut muncul dalam urutan atas daftar cipher alfabet tersebut akan lebih tinggi dari huruf lainnya.

Sekarang kita akan mencoba mengganti huruf yang sering muncul pertama dalam cipherteks, yaitu ‘j’ dengan huruf ‘e’, dan huruf ‘t’ tetap dengan huruf ‘t’. Kita mendapat bentuk baru sebagai berikut:

```
  -e ---- t-e-e t--t-- t- -e -e-- e---
e-t,
t--t --- -e- --e --e-te- e----, t--t
t-e-
--e e----e- -- t-e-- --e-t-- --t- -e-
t---
-----e----e ----t-,
t--t ----- t-e-e --e ---e,
---e-t- --- t-e ------t -- ------e--.
--- t--t t- -e---e t-e-e ----t-,
---e---e-t- --e ---t-t-te- ----- -e-,
-e------ t-e-- ---t ---e--
---- t-e ----e-t -- t-e ---e--e-.
```

Kita bisa mulai menebak kata-kata yang mungkin bisa kita prediksi dari sekarang. Tiga huruf dengan huruf ‘t’ di depan dan ‘e’ di belakang bisa jadi merupakan kata ‘the’. Itu berarti, asumsi kita kata ‘tej’ dalam cipherteks berkorespondensi dengan kata ‘the’ dalam plainteks, maka huruf ‘e’ dalam cipherteks berkorespondensi dengan huruf ‘h’ dalam plainteks. Jika kita coba substitusikan :

```
  -e h--- the-e t--th- t- -e -e-- e---
e-t,
th-t --- -e- --e --e-te- e----, th-t
the-
--e e----e- -- the-- --e-t-- --th -e-
t---
-----e----e ---ht-,
```

```
th-t ----- the-e --e ---e,
---e-t- --- the ------t -- h-----e--.
--- th-t t- -e---e the-e ---ht-,
---e---e-t- --e ---t-t-te- ----- -e-,
-e------ the-- ---t ---e--
---- the ----e-t -- the ---e--e-.
```

Jika kita substitusi ‘tewt’ dengan ‘that’, dan ‘ti’ dengan ‘to’, ‘ewnnvpjcc’ dengan ‘happiness’ kita dapatkan :

```
  -e ho-- these t--ths to -e se-- e-i-
ent,
that a-- -en a-e --eate- e--a-, that
the-
a-e en-o-e- -- thei- --eato- -ith -e-
tain
-na-iena--e -i-hts,
that a-on- these a-e -i-e,
-i-e-t- an- the p--s-it o- happiness.
an- that to se---e these -i-hts,
-o-e-n-ents a-e instit-te- a-on- -en,
-e-i-in- thei- --st po-e-s
--o- the -onsent o- the -o-e-ne-.
```

Dari sini kita sudah bisa melihat beberapa kata seperti `-onsent` bisa diisi menjadi ‘consent’, `-e-tain` menjadi ‘certain’, dan sebagainya. Setelah kita coba substitusi semua, maka nampak hasil plainteks hasil dekripsi yang kita lakukan:

```
  we hold these truths to be self
evident,
that all men are created equal, that
they
are endowed by their creator with
certain
unalienable rights,
that among these are life,
liberty and the pursuit of happiness.
and that to secure these rights,
governments are instituted among men,
deriving their just powers
from the consent of the governed.
```

III. DICTIONARY ATTACK

Dictionary attack merupakan sebuah serangan yang sering digunakan sebagai bagian dari *brute force attack* (serangan dengan mengujikan semua kemungkinan yang dapat membuahkan hasil) pada sebuah *password* suatu aplikasi.

Perbedaan utama *dictionary attack* dengan metode *brute force* biasa adalah, jika brute force mencoba semua kemungkinan dimulai dari kombinasi huruf yang paling sederhana, maka *dictionary attack* akan mencoba dari kombinasi koleksi kata-kata yang sudah disimpan sebelumnya (itulah sebabnya disebut *dictionary*).

Kombinasi kata-kata ini kemungkinan merupakan kata kunci *password*, sehingga waktu pencarian kata kunci akan lebih cepat daripada sekedar mengujikan semua kombinasi huruf. Dasar pemikiran untuk metode *dictionary attack* ini adalah, untuk menyimpan *password* suatu aplikasi, seseorang cenderung menggunakan kata-kata yang gampang diingat dan berasal dari kehidupan sehari-hari daripada kombinasi karakter yang acak.

Semakin banyak daftar kata-kata dalam *dictionary*, semakin besar pula peluang sukses dalam membongkar *password*.

IV. IMPLEMENTASI *DICTIONARY ATTACK* PADA SUBSTITUSI MONOALFABETIK

Seperti yang sudah dibahas pada bagian analisis frekuensi, pada metode substitusi monoalfabetik, setiap huruf akan disandikan dengan huruf lainnya. Perubahan huruf ini bisa dipastikan akan mengaburkan arti kata dari plainteks, tapi yang perlu diingat, pola dari kombinasi huruf akan tetap sama.

Berbeda dengan penggunaan *dictionary attack* pada *password*, dimana panjang *password* tidak diketahui, pada serangan terhadap substitusi monoalfabetik panjang teks diketahui. Hal ini memudahkan serangan, apalagi pola kombinasi huruf sudah diketahui.

Sebagai contoh, jika terdapat cipherteks denagn simbol-simbol sebagai berikut:

χηωνβζωυ

Asalkan kita tahu bahwa teks tersebut dalam bahasa Inggris dan menggunakan substitusi monoalfabetik, kita bisa langsung menebak kata tersebut dengan mencari dalam kamus kata bahasa Inggris dengan 8 huruf dan huruf ketiga dan ketujuh berulang sedangkan huruf lainnya saling berbeda. Kata-kata tersebut dalam bahasa Inggris antara lain:

ABORTION ABSTRUSE
ADOPTION AGENCIES
ANDROIDS ANTIDOTE
BANDYING BENCHING
BLEACHED BLEACHES
BOUTIQUE BREACHED
BREACHES BREASTED
.....

Sekarang kita akan mencoba cipherteks lainnya:

fbnl fzbfsw chwnbzwu

yang merupakan cipherteks dari plainteks “crop circle theories”.

Untuk pengujian dengan *tool dictionary attack*, kita akan menggunakan *tool* yang bisa langsung dipakai dari http://rumkin.com/tools/cipher/cryptogram-solver.php.

Dengan memasukkan cipherteks yang hendak kita pecahkan dengan pilihan mode *dictionary* English-Large., kita dapatkan hasil-hasil sebagai berikut:

PIKA PEIPUS BOSKIEST
ARID ABRAMO CHOIRBOY
ARIE ABRAMO CHOIRBOY
ARIL ABRAMO CHOIRBOY
ARIN ABRAMO CHOIRBOY
ARIZ ABRAMO CHOIRBOY
TUBA TRUTHS DISBURSE
GUTS GAUGER MORTUARY
PUTS PAUPER MORTUARY
PUTZ PAUPER MORTUARY
CROP CIRCLE THEORIES
CROW CIRCLE THEORIES

Dapat kita lihat bahwa meskipun *dictionary attack* dapat langsung memberikan hasil berupa kalimat-kalimat yang sesuai dengan pola pada cipherteks, hasil yang ditunjukkan masih banyak.

Jika kita coba dengan contoh cipherteks pada http://crypto.lkdev.com, kita ambil kalimat sebelum koma pertama:

MJ EIBO TEJCJ TSXTEC TI LJ CJBU JRVOJPT

Hasil yang kita dapat dari *dictionary attack* dengan mode *dictionary* English-small (karena hasil yang didapat dari mode yang lebih besar menghasilkan hasil yang terlalu banyak):

ME HOLD THESE TRUTHS TO BE SELF EVIDENT
WE HOLD THESE TRUTHS TO BE SELF EVIDENT
BE HOLD THESE TRUTHS TO ME SELF EVIDENT
WE HOLD THESE TRUTHS TO ME SELF EVIDENT
BE HOLD THESE TRUTHS TO WE SELF EVIDENT
ME HOLD THESE TRUTHS TO WE SELF EVIDENT

Hasil yang didapat masih lebih dari satu. Jika kita coba dengan kalimat sampai sebelum tanda koma kedua:

MJ EIBO TEJCJ TSXTEC TI LJ CJBU JRVOJPT TEWT WBB KJP WSJ ASJWTJO JQXWB

ternyata hasilnya masih lebih dari satu. Bahkan ketika panjang kalimat sampai sebelum tanda koma ketiga:

MJ EIBO TEJCJ TSXTEC TI LJ CJBU JRVOJPT, TEWT WBB KJP WSJ ASJWTJO JQXWB, TEWT TEJY WSJ JPOIMJO LY TEJVS ASJWTIS MVTE AJSTWVP XPWBVJPWLBJ SVGETC

pada mode *dictionary* English-small dan medium tidak mendapatkan hasil. Pada mode *dictionary* English-large barulah mendapatkan hasil sebagai berikut:

WE HOLD THESE TRUTHS TO BE SELF EVIDENT THAT ALL JEN ARE CREATED EQUAL THAT THEY ARE ENDOWED BY THEIR CREATOR WITH CERTAIN UNALIENABLE RIGHTS

WE HOLD THESE TRUTHS TO BE SELF EVIDENT THAT ALL KEN ARE CREATED EQUAL THAT THEY ARE ENDOWED BY THEIR CREATOR WITH CERTAIN UNALIENABLE RIGHTS

WE HOLD THESE TRUTHS TO BE SELF EVIDENT THAT ALL MEN ARE CREATED EQUAL THAT THEY ARE ENDOWED BY THEIR CREATOR WITH CERTAIN UNALIENABLE RIGHTS

WE HOLD THESE TRUTHS TO BE SELF EVIDENT THAT ALL PEN ARE CREATED EQUAL THAT THEY ARE ENDOWED BY THEIR CREATOR WITH CERTAIN UNALIENABLE RIGHTS

WE HOLD THESE TRUTHS TO BE SELF EVIDENT THAT ALL ZEN ARE CREATED EQUAL THAT THEY ARE ENDOWED BY THEIR CREATOR WITH CERTAIN UNALIENABLE RIGHTS

Terlihat bahwa perbedaan hanya terlihat pada pendeskripsian pada bagian ‘men’ dimana hasil *dictionary attack* memberikan hasil saran ‘jen’, ‘ken’, ‘pen’, dan ‘zen’ selain ‘men’.

V. Analisis Implementasi

Dari hasil percobaan serangan *dictionary attack* di atas dapat terlihat bahwa hasil dari pencocokan pola pada cipherteks dengan kata-kata dari kamus tidak menghasilkan satu hasil saja, melainkan banyak kata/ kalimat yang sesuai dengan pola kombinasi simbol pada cipherteks.

Semakin panjang teks, kemungkinan akan menghasilkan jumlah pencarian yang semakin sempit. Hal ini bisa dipahami karena semakin panjang teks, kemungkinan macam huruf yang digunakan akan semakin banyak. Semakin banyak jenis huruf yang dipakai dalam teks, maka keragaman kombinasi kata akan makin banyak dan makin mempersempit pencarian oleh *dictionary attack* pada kamus yang bersesuaian.

Lalu bagaimana jika panjang teks tidak cukup untuk menghasilkan hasil pencarian yang sedikit? Pada contoh di atas, bahkan cipherteks sepanjang tiga kalimat masih menghasilkan lima hasil yang sesuai dengan pola cipherteks.

Jawaban sederhananya adalah apakah hasil yang ditampilkan memiliki makna atau tidak. Percakapan dalam setiap bahasa memiliki suatu aturan atau struktur yang

membuat suatu kalimat atau rangkaian kata memiliki makna, yaitu tata bahasa. Pada contoh di atas, cipherteks 'MJ EIBO TEJCJ TSXTEC TI LJ CJBU JRVOJPT' menghasilkan enam hasil yang kesemuanya memiliki kemiripan arti. Tapi dengan memahami tata bahasa Inggris, kita bisa menebak bahwa 'WE HOLD THESE TRUTHS TO BE SELF EVIDENT' lebih masuk akal sebagai plainteks dari cipherteks tersebut. Pada contoh 'fbnl fzbfsw chwnbzwu', plainteks yang paling mungkin adalah antara 'CROP CIRCLE THEORIES' atau 'CROW CIRCLE THEORIES'. Tapi dengan memahami bahwa istilah 'crop circle' lebih umum daripada 'crow circle', kita bisa menebak bahwa plainteks tersebut adalah 'crop circle theories'.

Pada contoh yang lebih awal, 'χηωνβζωυ' menghasilkan hasil yang terlampau banyak, dan karena cipherteks tersebut hanya merupakan satu kata, kita tidak bisa menentukan mana hasil yang merupakan plainteks dari cipherteks tersebut, karena hasil yang ditampilkan merupakan sebuah kata yang tidak tersangkut-paut dengan tata bahasa. Jika hasil seperti ini, kita hanya bisa menebak-nebak kata apa kira-kira yang disandikan, dengan melihat konteks dari situasi pembuatan cipherteks dan memilah kata mana yang mungkin digunakan dalam situasi tersebut.

Oleh karena kriptografi dengan substitusi monoalfabetik dapat dengan mudah dipecahkan berdasarkan pola kalimat, pembuatan cipherteks dengan substitusi monoalfabetik tidak dengan mentah disubstitusi dari plainteks awalnya, melainkan diproses dengan berbagai metode agar membingungkan kriptanalis dalam memecahkan cipherteks.

Salah satu metode yang umum digunakan adalah menghilangkan spasi pada cipherteks, sehingga cipherteks akan berupa satu baris saja. Jika seperti ini, *dictionary attack* akan mengalami kesulitan dalam memecahkan cipherteks. Hal ini dikarenakan tidak diketahuinya panjang kata awal, kata kedua, dan seterusnya, sehingga aplikasi tidak akan bisa mencoba mencocokkan pola kata pada cipherteks dengan kata dari kamus.

Salah satu cara penanganan kasus ini adalah dengan menambahkan pemrosesan pada *dictionary attack* yang mencoba memecah-mecah barisan huruf menjadi sejumlah huruf yang masih memiliki pola pada kamus. Pemecahan bisa dimulai dengan sejumlah kecil kata yang masih mungkin memiliki makna, termasuk artikel yang hanya terdiri dari sedikit huruf.

fbnlfzbfswchwnbzwu
f bnlfzbfswchwnbzwu
fb nlfzbfswchwnbzwu
fbn lfzbfswchwnbzwu
....
fbnl f zbfswchwnbzwu
fbnl fz bfswchwnbzwu
fbnl fzb fswchwnbzwu

Setiap kombinasi yang dapat menghasilkan (yaitu dapat membentuk kata-kata sesuai dengan kamus) akan didaftar. Jika terdapat banyak hasil, pengecekan pada tata bahasa bisa digunakan untuk memilah mana hasil yang paling tepat sebagai plainteks.

Untuk teks yang panjang jelas akan sangat menghabiskan waktu dengan cara pengecekan seperti ini. Salah satu alternatif solusi adalah mengambil penggalan sebagian teks, dan mencoba mencari kalimat yang sesuai tata bahasa dalam penggalan ini. Dari kalimat yang ditemukan, kita bisa membuat daftar substitusi antara cipher alfabet dengan plain alfabetnya untuk membantu mendekripsi teks yang lebih panjang. Tentu saja akan ada tambahan penanganan khusus, karena penggalan yang kita ambil bisa saja memotong suatu kata sehingga potongan kata tersebut menjadi kata tidak berarti yang tidak terdapat di kamus.

Kelemahan utama *dictionary attack* adalah jika terdapat penggunaan kata pada plainteks yang tidak terdapat pada kamus. Akibatnya pemrosesan terhadap cipherteks secara keseluruhan akan gagal. Kita sudah membuktikannya pada contoh cipherteks 'MJ EIBO TEJCJ TSXTEC TI LJ CJBU JRVOJPT, TEWT WBB KJP WSJ ASJWTJO JQXWB, TEWT TEJY WSJ JPOIMJO LY TEJVS ASJWTIS MVTE AJSTWVP XPWBVJPWLBJ SVGETC', dimana mode *dictionary* English-small dan medium gagal membuahkan hasil dikarenakan tidak memiliki kata 'UNALIENABLE' dalam kamusnya.

Begitu pula jika terdapat penggunaan nama-nama atau istilah-istilah yang tidak lazim dalam cipherteks. Kemampuan manusia dalam mempersepsi kata tanpa harus melihat detil semua huruf juga salah satu hal yang dimanfaatkan para kriptografer dalam menyandikan pesan mereka. Sebagai contoh, kata 'pljrn' bisa kita persepsikan sebagai 'pelajaran'. Pemakaian kata-kata yang tidak baku atau istilah-istilah slank juga menjadi hambatan dalam pemecahan melalui *dictionary attack*. Salah satu solusi nya antara lain memperbesar perbendaharaan kata dalam kamus, meskipun nantinya jumlah hasil pemecahan dari suatu cipherteks juga akan makin besar.

Dalam hal ini, teknik frekuensi analisis masih lebih unggul karena dalam mencoba pemecahan huruf per huruf, ada kemungkinan kriptanalisis menemui kata yang masih dalam taraf umum dan bisa dikenali.

VI. KESIMPULAN

Teknik serangan *dictionary attack* pada kriptografi berbasis substitusi monoalfabetik merupakan teknik yang sangat bergantung pada pola huruf dan kesesuaian pola pada cipherteks dengan koleksi kata-kata dalam kamus. Semakin besar perbendaharaan kata kamus, semakin besar pula kemungkinan memecahkan cipherteks.

Teknik ini sangat lemah terhadap adanya istilah yang tidak dikenali dalam cipherteks karena dengan adanya satu istilah saja yang tidak terdapat dalam kamus, maka

seluruh kerja *dictionary attack* akan gagal. Hal ini sebenarnya masih bisa ditangani dengan penambahan penanganan kasus khusus pada algoritma *dictionary attack* seperti pengecekan tata bahasa, pemotongan teks panjang menjadi lebih kecil, dan lainnya.

Teknik *dictionary attack* menunjukkan bahwa pada metode kriptografi seperti apapun, pola pada cipherteks selalu ada. Selama hal yang dituliskan pada plainteks merupakan sesuatu yang bermakna dan mengikuti suatu kaidah tata bahasa tertentu, pola pada cipherteks juga akan selalu muncul. Hal ini yang dimanfaatkan *dictionary attack* dalam serangannya terhadap cipherteks.

Algoritma kriptografi yang canggih tidak akan pernah menghilangkan pola plainteks dalam cipherteks. Semakin canggih algoritma kriptografi hanya semakin memperkabur pola dalam cipherteks agar tidak mudah dibaca oleh kriptanalis.

REFERENCES

[1] http://www.simonsingh.net/The_Black_Chamber/kamasutra.html access date: 20-03-2011

[2] H. Lee Kwang, Basic Encryption and Decryption, Department of Electrical Engineering & Computer Science, KAIST, mars 2000

[3] Ibrahim A. Al-Kadi "The origins of cryptology: The Arab contributions", Cryptologia, 16(2) (April 1992) pp. 97–126.

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 22 Maret 2011

ttd

Haryus Aminul Akbar 13507016