STUDI MENGENAI *CUBE ATTACK*

Firdi Mulia – NIM : 13507045

Program Studi Teknik Informatika, Institut Teknologi Bandung
Jl. Ganesha 10, Bandung
E-mail : if17045@students.if.itb.ac.id

ABSTRAK

Makalah ini membahas tentang studi *Cube Attack* , salah satu metode kriptanalisis yang ditemukan oleh Itai Dinur dan Adi Shamir dan dapat diterapkan dalam banyak variasi algoritma kunci simetri. untuk menyandikan data yang disimpan dalam media penyimpanan. Metode ini termasuk baru karena diumumkan pada bulan September 2008 dalam bentuk cetakan dan direvisi yang diletakkan secara *online* pada bulan Januari 2009. Bit cipherteks y_i, yang dihasilkan algoritma ini adalah nilai polinom yang tergantung dari kunci publik $v_1,\ldots,v_m$ menjadi bit dari plainteks dari blok cipher atau bit dari vektor inisial untuk stream cipher dan bergantung pada variabel privat x1,…,xn yang menjadi bit kunci. Serangan ini adalah *known-plaintext attack* dan mempunyai dua tahapan. Pada tahapan pertama tujuan utamanya adalah menemukan sebuah fungsi yang bergantung pada bit kunci dan menemukan kasus-kasus dimana fungsi tersebut adalah fungsi linear dan merekonstruksinya. Pada tahapan selanjutnya yaitu tahapan *on line*, penyerang menyelesaikan persamaan linear untuk menemukan beberapa bit dari kunci. Secara umum, bentuk eksplisit dari polinom tidak diketahui dan cipher bisa saja berupa *black box*, tetapi *cube attack* masih dapat diterapkan.

Menurut Dinur dan Shamir, mereka sudah mempraktekkan metode ini pada sebuah stream cipher dan hasilnya adalah metode ini paling efektif dibandingkan dengan metode-metode sebelumnya. Penemu ini juga mengklaim kalau sebuah penyerangan pada Trivium (salah satu metode enkripsi yang rumit) direduksi sampai 735 kali inisialisasi dengan kompleksitas 2^{30} bit operasi dan bahkan bisa dikembangkan untuk memecahkan 1100 dari 1152 inisialisasi Trivium. Dan memang terbukti pada bulan Desember 2008, metode ini diakui sebagai metode penyerangan terbaik terhadap Trivium.

Kemudian penyerangan ini juga dicoba pada 4 putaran blok CTC (*Courtois Toy Cipher*) untuk versi dengan 120 bit blok dan 255 bit kunci dan hasilnya didapatkan semua nilai dari bit kunci dengan kompleksitas yang jauh lebih kecil dibandingkan *exhaustive search*.

***Kata kunci**: cube attack*, *known-plaintext attack* , *trivium*, *courtois toy cipher*, *exhaustive search*, fungsi linear, polinom, enkripsi, dekripsi.

1. PENDAHULUAN

Pengiriman data dan penyimpanan data melalui media elektronik memerlukan suatu proses yang dapat menjamin keamanan dan keutuhan dari data yang dikirimkan tersebut. Data tersebut harus tetap rahasia selama pengiriman dan harus tetap utuh pada saat penerimaan di tujuan. Untuk memenuhi hal tersebut, dilakukan proses penyandian (enkripsi dan dekripsi) terhadap data yang akan dikirimkan. Enkripsi dilakukan pada saat pengiriman dengan cara mengubah data asli menjadi data rahasia sedangkan dekripsi dilakukan pada saat penerimaan dengan cara mengubah data rahasia menjadi data asli. Jadi data yang dikirimkan selama proses pengiriman adalah data rahasia, sehingga data asli tidak dapat diketahui oleh pihak yang tidak berkepentingan. Data asli hanya dapat diketahui oleh penerima dengan menggunakan kunci rahasia.

Tetapi karena sifat manusia yang ingin tahu rahasia seseorang, maka berkembanglah teknik-teknik untuk mendekripsi suatu data yang dienkripsi tanpa mengetahui kunci rahasianya yang disebut kriptanalisis.

Kriptanalisis pertama yang tercatat dilakukan oleh orang Arab bernama Al-Kindi yang melakukan kriptanalisis dengan analisa frekuensi yang kemudian metode ini menjadi dasar dari kriptanalisis terhadap cipher klasik.

Gambar 1. Halaman pertama dari buku *Manuscript on Deciphering Cryptographic Messages* karangan Al-Kindi

Seiring berjalannya waktu, algoritma enkripsi berkembang dan begitu juga dengan kriptanalisis. Salah satu jenis algoritma enkripsi adalah algoritma kunci simetrik. Algoritma kunci simetrik yang sudah ada saat ini dan diterapkan ada banyak sehingga sangat mengundang kebanyakan orang untuk menyerang algoritma kunci simetrik dan membuat kriptanalisisnya. Salah satunya metode kriptanalisis yang bisa diterapkan pada banyak variasi algoritma kunci simetrik adalah *Cube Attack*.

2. KRIPTANALISIS

2.1 Definisi Kriptanalisis

Kriptanalisis berasal dari bahasa Yunani *kryptos* yang berarti tersembunyi dan *analyein* yang berarti melepaskan. Kriptanalisis berarti sebuah studi mengenai metode untuk mendapatkan arti dari informasi yang terenkripsi, tanpa memiliki kunci akses ke informasi rahasia tersebut. Studi ini melibatkan pengetahuan mengenai bagaimana sistem bekerja dan menemukan sebuah kunci rahasia.

Metode kriptanalisis terus berkembang dan semakin kompleks, dari penggunaan pena dan kertas di masa lalu, sekarang menggunakan skema berbasis komputer.

2.2 Syarat Kriptanalisis

Kriptanalisis bisa dilakukan dengan sejumlah asumsi tentang berapa banyak akses yang dipunyai penyerang pada suatu sistem. Contoh asumsi tersebut adalah :

- Ciphertext-only
  Kriptanalis mempunyai akses hanya pada sekumpulan cipherteks atau kode teks.
- Known-plaintext
  Penyerang memiliki sejumlah cipherteks dimana dia tahu plainteks yang berkorespondensi dengan cipherteks tersebut.
- Chosen-plaintext
  Penyerang bisa mendapatkan plainteks yang berkorespondensi pada sejumlah cipherteks yang bisa dia pilih.
- Adaptive chosen-plaintext
  Serupa dengan chosen-plaintext, kecuali penyerang dapat memilih plainteks berdasarkan informasi yang didapatkan dari enkripsi sebelumnya.
- Related-key attack
  Mirip dengan chosen-plaintext attack, hanya saja penyerang bisa mendapatkan cipherteks yang terenkripsi dengan dua kunci yang berbeda. Kuncinya tidak diketahui, teteapi relasi diantaranya diketahui. Sebagai contoh, dua kunci hanya berbeda pada satu bit.

2.3 Sumber Daya Komputasional yang Dibutuhkan

Penyerangan dapat dicirikan dengan sumber daya yang dibutuhkna. Sumber daya tersebut meliputi :

- Waktu
  Jumlah langkah komputasi (seperti enkripsi) yang harus dilakukan.
- Memori
  Jumlah penyimpanan yang diperlukan untuk melakukan serangan.
- Data
  Jumlah dari plainteks dan cipherteks yang diperlukan.

Kadang-kadang sulit untuk memprediksi kuantitas ini secara tepat, khususnya ketiak serangan tidak benar-benar diimplementasikan untuk dicoba. Tetapi kebanyakan kripatanalis setidaknya menyediakan perkiraan kesulitan dari serangan mereka.

3. METODE CUBE ATTACK

3.1 Latar Belakang Matematika

Pada bagian ini, variabel privat dan publik tidak perlu dibedakan. Misalnya p adalah sebuah polinom dari n variabel x1,…,xn dari bidang GF(2). Untuk sebuah subset dari indeks I = {i1,…,ik} merupakan elemen dari {1,…,n}. Misalnya terdapat monomial t1 = xi1, …, xix. Kemudian terdapat sebuah dekomposisi

$$\boldsymbol{p(x_1,\dots,x_n) = t_1.p_{s(I)} + q(x_1,\dots,x_n)}$$

Dimana polinom $p_{s(I)}$ tidak bergantung dari variabel $x_{i1},\dots,x_{ik}$.

Contohnya:
Misalnya ada sebuah polinom p dengan derajat 3 yang bergantung pada 5 variabel:

$$\boldsymbol{p(x_1,x_2,x_3,x_4,x_5) = x_1x_2x_3 + x_1x_2x_4 + x_2x_4x_5 + x_1x_2 + x_3x_5 + x_2 + x_5 = 1}$$

Misalkan I = {1.2} adalah subset dari indeks. Maka polinom P dapat didekomposisi menjadi :

$$\boldsymbol{p(x_1,x_2,x_3,x_4,x_5) = x_1x_2(x_3 + x_4 + 1) + (x_2x_4x_5 + x_3x_5 + x_2 + x_5 + 1)}$$

Dengan menggunakan notasi diatas:

$$t_1 = x_1x_2,$$
$$p_{s(I)} = x_3 + x_4 + 1,$$
$$q(x_1,x_2,x_3,x_4,x_5) = x_2x_4x_5 + x_3x_5 + x_2 + x_5 + 1$$

Derajat tertinggi dari polinom p yang dapat disebut monom t1, sehingga

$$\text{Deg}(p_{s(I)})=1,$$

Yang berarti polinom ps(I) berkorespondensi dengan subset dari indeks I adalah linear, yang bukan merupakan sebuah konstanta.

Misalnya I = {i1,…,Ik} komplemen dari {1,…,n} merupakan subset yang tetap dari k indeks.
I mendefinisikan k dimensi boolean cube C, dimana pada setiap tempat dari setiap indeks diletakkan 0 atau 1. Sebuah vektor v anggota C1 menurunkan polinom pv yang bergantung pada n-k variabel, dimana pada polinom dasar p, kita meletakkan nilai berkorespondensi dengan vektor v. Menjumlahkan semua vector dalam cube C1 bisa didapatkan polinom :

$$P_i = \sum_{v \epsilon C1} Pv$$

Misalkan p adalah sebuah polinom dari bidang GF(2) dan I komplemen {1,…,n} dari subset indeks. Maka bisa didapatkan:

$$p_I = p_{s(I)} \bmod 2$$

3.2 Struktur Serangan

Misalnya ada cryptosystem yang digambarkan dengan polinom :

$$p(v_1,\dots,v_m,x_1,\dots,x_n)$$

yang bergantung pada *m* variabel publik *v*1,…,*vm* (nilai awal atau plainteks) dan pada n variabel privat x1,…,xn (kuncinya). Nilai dari polinom merepresentasikan bit cipherteks. Secara umum, polinom p tidak secara eksplisit diketahui; bisa berupa sebuah *black box*. Pembahasan akan berfokus pada known plaintext attack, dimana tahap pada preproses penyerang juga mempunyai akses ke variabel privat (nilai awal atau kunci).
Secara umum, tahapan penyerangan terbagi menjadi dua yaitu :

1. Tahap preproses
   Penyerang dapat mengubah nilai dari variabel public dan privat. Tujuan dari tahap ini adalah untuk mendapatkan sebuah sistem dari persamaan linear dari variabel privat.
2. Tahap *on-line* dari serangan
   Pada tahapan ini, kunci menjadi rahasia. Penyerang dapat mengubah nilai dari variabel public. Dia menambah bit keluaran, dimana input berjalan pada beberapa cube multi dimensi. Tujuan dari tahap ini adalah mendapatkan ruas kanan dari persamaan linear. Sistem dari persamaan linear ini dapat diselesaikan dengan diketahui beberapa bit dari kunci.

3.2.1 Tahap preproses

Pada tahap ini terdiri dari 3 langkah:

1. Langkah pertama adalah adalah untuk menetapkan dimensi dari cube dan variabel publik dimana kedua variabel tersebut akan dijumlahkan. Kedua variabel tersebut disebut sebagai variabel *tweakable*, dan variabel publik lainnya diset menjadi nol. Pada kasus tersebut bisa diketahui kalau derajat d dari polinom dasar, dan dimesi dari cube diset menjadi d-1.
2. Langkah kedua dilakukan *summation* dari sebuah cube tetap untuk beberapa nilai dari variabel privat dan mengumpulkan nilai yang didapatkan.
3. Dilakukan test linear untuk fungsi yang didapatkan dari variabel privat dan menyimpannya ketika sudah linear.

   $$f(x \oplus x') = f(x) \oplus f(x') \oplus f(0),$$

   Dimana x = $(x_1,\dots,x_n)$ adalah variabel privat (kunci)

Tugas berikutnya adalah menghitung bentuk pasti (koefisien) dari fungsi linear yang didapat dari variabel privat. Bentuk bebas dari fungsi linear yang kita dapatkan menempatkan semua argument sama dengan nol. Koefisien dari variabel x_i sama dengan 1 jika dan hanya jika perubahan nilai dari variabel ini mengimplikasikan perubahan nilai dari fungsi. Koefisien dari variabel x_i sama dengan 0 jika dan hanya jika perubahan dari variabel ini tidak mengimplikasikan perubahan nilai dari fungsi tersebut.

Tujuan dari tahap dari serangan ini adalah untuk mengumpulkan sebanyak mungkin persamaan linear yang independen. Persamaan-persamaan tersebut membentuk sistem persamaan linear dari variabel kunci. Sistem persamaan linear ini akan digunakan dalam tahap on line. Tahap preproses ini hanya dilakukan sekali dalam kriptanalisis pada algoritma ini.

3.2.2 Tahapan *on-line*

Sekarang penyerang mempunyai akses hanya pada variabel public (plaintext untuk block cipher, nilai awal untuk stream cipher), dimana penyerang dapat mengubah dan menghitung bit yang berhubungan dari ciphertext dalam nilai variabel privat yang tidak diketahui.

Tujuan dari serangan ini adalah menemukan beberapa bit dari variabel privat dengan kompleksitas, dimana akan lebih rendah dibandingkan *exhaustive search* dalam serangan *brute force*.

Penyerang menggunakan sistem persamaan linear yang diturunkan untuk variabel kunci (bit-bit yang tidak diketahui dari kunci), dimana ruas kanan dari persamaan ini adalah nilai bit dari cipherteks yang didapatkan setelah *summation* dalam cube yang sama dalam tahap preproses.

Serangan cube bisa digunakan pada cipher simetrik dimana polinom-polinom yang ada menggambarkan sistem mempunyai derajat yang rendah. Kemudian penyerang dapat menemukan beberapa bit yang tersisa dari kunci yang bisa ditemukan dengan pencarian *brute force*. Setelah tahapan preproses yang berhasil, tahapan on line dari serangan dapat dilakukan beberapa kali untuk beberapa kunci tidak diketahui yang berbeda.

Serangan cube dapat diaplikasikan, secara umum, terhadap sistem kripto tanpa mengetahui struktur dalamnya. Penyerang harus menyelesaikan tahapan preproses dengan benar dan pada tahapan on line memiliki akses terhadap implementasi dari algoritma (untuk melakukan *summation* dari cube pada kunci yang tidak diketahui).

4. PENGAPLIKASIAN SERANGAN CUBE PADA ALGORITMA ENKRIPSI TRIVIUM

Algoritma Trivium yang ditemukan oleh C. de Canniere dan B. Preneel adalah salah satu finalis dari kompetisi e-stream. Parameter dasarnya adalah 80-bit kunci dan 80-bit nilai awal.

State dari trivium adalah 288 bit dimasukkan ke dalam 3 register nonlinear dengan panjang yang berbeda. Pada setiap putaran dari algoritma, register digeser sebanyak satu bit. Umpan balik dari setiap register diberikan oleh fungsi nonlinear.

$$(s1, s2, \ldots, s93) \leftarrow (k1, k2, \ldots, k80, 0, \ldots, 0)$$
$$(s94, s95, \ldots, s177) \leftarrow (IV1, IV2, \ldots, IV80, 0, \ldots, 0)$$
$$(s178, s179, \ldots, s288) \leftarrow (0, 0, \ldots, 0, 1, 1, 1)$$

Untuk i=1 sampai 1152

$$t1 \leftarrow s66 + s93$$
$$t2 \leftarrow s162 + s177$$
$$t3 \leftarrow s243 + s288$$
$$t1 \leftarrow t1 + s91 \cdot s92 + s171$$
$$t2 \leftarrow t2 + s175 \cdot s176 + s264$$
$$t3 \leftarrow t3 + s286 \cdot s287 + s69$$
$$(s1, s2, \ldots, s93) \leftarrow (t3, s1, \ldots, s92)$$
$$(s94, s95, \ldots, s177) \leftarrow (t1, s94, \ldots, s176)$$
$$(s178, s179, \ldots, s288) \leftarrow (t2, s178, \ldots, s287)$$

Penentuan string keluaran (z_i) dari panjang maksimal sampai $N = 2^{64}$ bit, dapat direpresentasikan sebagai :

$$t1 \leftarrow s66 + s93$$
$$t2 \leftarrow s162 + s177$$
$$t3 \leftarrow s243 + s288$$
$$zi \leftarrow t1 + t2 + t3$$
$$t1 \leftarrow t1 + s91 \cdot s92 + s171$$
$$t2 \leftarrow t2 + s175 \cdot s176 + s264$$
$$t3 \leftarrow t3 + s286 \cdot s287 + s69$$
$$(s1, s2, \ldots, s93) \leftarrow (t3, s1, \ldots, s92)$$
$$(s94, s95, \ldots, s177) \leftarrow (t1, s94, \ldots, s176)$$
$$(s178, s179, \ldots, s288) \leftarrow (t2, s178, \ldots, s287)$$

Dinur dan Shamir menginvestigasi versi reduksi dari Trivium yang berisi 672 (dari 1152) putaran inisialisasi. Selama tahap preproses, mereka mendapatkan 63 persmaaan independen linear yang berkorespondensi dengan cube 12 dimensi dan indeks bit keluaran dari 672 sampai 685.

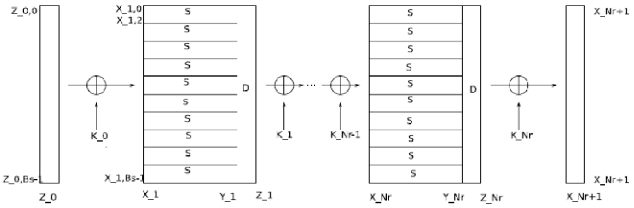
Pada tahapan on line, penyerang harus menemukan nilai dari persamaan linear dengan menjumlahkan 63 cube 12 dimensi. Setelah menyelesaikan sistem persamaan linear, penyerang mendapatkan 63 bit dari kunci dan sisa 17 bit dari kunci ditemukan dengan pencarian *brute force*. Kompleksitas dari penyerangan ini adalah 2^{19} evaluasi dari algoritma yang diserang. Ini sangat kecil dibandingkan kompleksitas 2^{55} dari penyerangan metode lain pada Trivium versi ini.

5. PENGETESAN CUBE ATTACK DENGAN CTC

CTC (Courtois Toy Cipher) didesain oleh Nicolais Courtois untuk mengaplikasikan dan mengetes metode analisis aljabar. CTC ini merupakan sebuah jaringan SPN yang tidak terlalu besar dalam jumlah putaran, blok dan ukuran kunci.

Setiap putaran melakukan operasi yang sama dari data masukan, kecuali sebuah putaran kunci yang berbeda ditambahkan setiap putaran. Jumlah putaran dinotasikan dengan Nr. Keluaran dari putaran i - 1 adalah masukan untuk putaran i.

Setiap putaran terdiri dari aplikasi parallel dari B S-box (*S*), aplikasi dari *linear diffusion layer* (*D*), dan sebuah penambahan kunci terakhir dari putaran kunci. Kunci putaran K_0 ditambahkan dalam blok plainteks sebelum putaran pertama.



Gambar 2 Bagan CTC untuk B = 10

Bit plainteks $p_0 \ldots p_{Bs-1}$ diidentifikasikan dengan $Z_{0,0} \ldots Z_{0,Bs-1}$ dan bit cipherteks $c_0 \ldots c_{Bs-1}$ diidentifikasikan sebagai $x_{Nr+1,0} \ldots x_{nr+1,Bs-1}$ untuk notasi yang seragam.

S-box ditentukan dari permutasi

[7, 6, 0, 4, 2, 5, 1, 3]

Yang berarti ada $2^3 = 8$ masukan dan 8 keluaran. Bit keluaran adalah fungsi Boolean kuadrat dari bit masukan. Bentuk eksplisit dari fungsi ini tidak digunakan dalam serangan cube.

Layer difusi (D) didefinisikan sebagai

$$Z_{i,257 \bmod Bs} = Y_{i,0}$$

untuk semua i=1…Nr,

$$Z_{i,\,(1987j+257) \bmod Bs} = Y_{i,j} + Y_{i,(j+137) \bmod Bs}$$

untuk j tidak sama dengan 0 dan semua i, dimana Yi,j merepresentasikan bot input dan Zi,j merepresentasikan bit keluaran.

Jadwal kunci adalah sebuah permutasi sederhana dari bit:

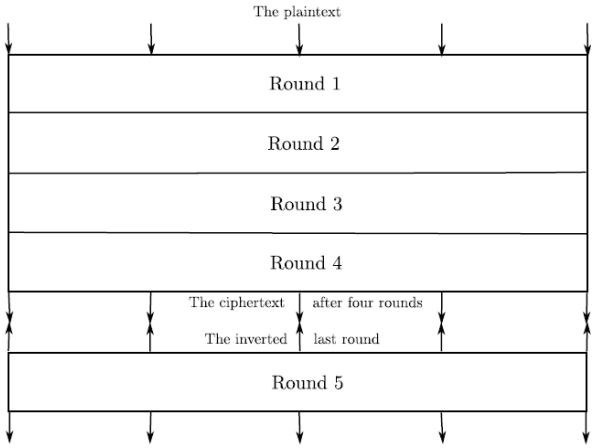
$$K_{i,j} = K_{0,(i+J) \bmod Bs}$$

Untuk semua I dan j, dimana K_0 adalah kunci utama.

Penambahan kunci dilakukan secara bit-wise:

$$X_{i+1,j} = Z_{i,j} + K_{i,j}$$

Untuk semua i = 1…Nr dan j = 1…Bs – 1, dimana Zi,j merepresentasikan bit keluaran dari layer difusi sebelumnya, $X_{i+1,j}$ bit masukan dari putaran berikutnya dan Ki,j merupakan bit dari kunci putaran sekarang.



Gambar 3 Skema Cube Attack pada CTC

Saya telah mencoba mengaplikasikan cube attack ke CTC dengan 4 putaran dan B = 40 dari S-box, dimana blok dan ukuran kunci adalah 120 bit.

Pada tahap preproses, dilakukan *summation* dari 50000 4 dimensi cube yang telah dipilih secara random dari plaintext (bit lainnya dari plainteks diset menjadi nol). Kemudian dari 757 box dihasilkan ekspresi linear untuk bit dari kunci. Untuk penurunan dari tiap ekspresi linear, digunakan 5000 test linear. Kemudian dipilih 120 ekspresi linear yang kemudian bisa memberikan sistem persamaan linear yang bisa diselesaikan untuk menemukan kunci.
Karena banyaknya ekspresi (lebih dari 120 ekspresi), saya hanya mencoba menghitung 10 persamaan sebagai contoh seperti yang ditunjukkan pada tabel dibawah.

Tabel 1 Contoh persamaan linear dari percobaan

Persamaan	Indeks Cube
1+x66+x68=c66	{4,5,22,52}
x27+x28=c105	{1,60,62,90}
1+x58=c18	{16,17,60,110}
1+x14=c80	{29,38,61,106}
1+x54+x56=c36	{41,55,64,115}
1+x115+x116=c66	{5,10,22,89}
1+x43=c11	{73,74,75,118}
x69+x70=c28	{58,68,77,110}
1+x25=c70	{73,76,93,104}
x78+x79=c114	{10,39,70,118}

Ada 120 persamaan linear independen seperti di atas untuk bit $x_0,\ldots,x_{119}$ untuk kunci: $c_0,\ldots,c_{119}$.

Pada tahap on line dari penyerangan, kita perlu menjumlahkan 120 cube yang dipilih (dimana kunci tidak dikethaui) untuk menemukan ruas kanan yang tepat untuk persamaan linear.

Kompleksitas dari serangan ini adalah sekitar $2^7.2^4 = 2^{11}$ enkripsi dari 4 putaran CTC utnuk menemukan 120-bit kunci. Faktanya, putaran cipher yang direduksi ini tampak pada sebuah sistem persamaan linear. Menurut pendapat saya, mungkin untuk putaran yang lebih banyak dari 4, akan mustahil untuk menemukan gambaran tersebut.

6. KESIMPULAN

Kesimpulan yang dapat dimbil dari studi *Cube Attack* ini adalah:

1. *Cube Attack* merupakan salah satu metode kriptanalisis yang efektif terhadap algoritma kunci simetrik.
2. Permasalahan utama dari *Cube Attack* adalah menemukan ekspresi linear untuk bit kunci pada tahap preproses.
3. Diperlukan kakas test linear dan heuristic untuk menentukan derajat aljabar dari polinom yang terlibat.
4. Dengan mengetahui metode *Cube Attack*, kita bisa mendesain algoritma enkripsi yang tidak bisa ditembus oleh metode kriptanalisis ini sehingga algoritma enkripsi tersebut akan lebih aman.

REFERENSI

[1] Cryptanalysis. *http://en.wikipedia.org/wiki/Cryptanalysis*. Tanggal akses : 22 Maret 2010.

[2] Dinur, Itai. Shamir, Adi. (2009). Side Channel Cube Attacks on Block Cipher. http://eprint.iacr.org/2009/127.pdf. Tanggal akses : 23 Maret 2010.

[3] Mroczkowski, Piotr. (2009). Cube Attack.on Courtois Toy Cipher. *http://eprint.iacr.org/2009/497.pdf*. Tanggal akses: 23 Maret 2010.

[4] Mroczkowski, Piotr. (2009). The Cube Attack. *http://events.iaik.tugraz.at/weworc09/9aa510c7c7aab1/abstracts/05.pdf*. Tanggal akses: 23 Maret 2010.

[4] Munir, Rinaldi. (2004). Bahan Kuliah IF5054 Kriptografi. Departemen Teknik Informatika, Institut Teknologi Bandung.

[5] Szmidt, Janusz. (2009). Application of Cube Attack to Block and Stream Cipher. *http://conf.fme.vutbr.cz/cecc09/lectures/szmidt.pdf*. Tanggal akses: 20 Maret 2010.