STUDI DAN MODIFIKASI ALGORITMA BLOCK CHIPER MODE ECB DALAM PENGAMANAN SISTEM BASIS DATA

Arief Latu Suseno – NIM: 13505019

Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika Institut Teknologi Bandung
Jl. Ganesha 10, Bandung
E-mail : if15019@students.if.itb.ac.id

Abstrak

Makalah ini menjelaskan pengamanan transmisi data hasil query basis data dengan menggunakan algoritma blok chiper mode operasi *electronic code book* (ECB) yang telah dimodifikasi. Pengamanan transmisi data dilakukan dengan mengenkripsi data hasil *query* yang dikirimkan oleh sistem manajemen basis data untuk kemudian didekripsi oleh aplikasi yang berjalan di sisi pengguna.

Mode ECB yang digunakan adalah mode ECB yang telah dimodifikasi dengan tujuan untuk mengatasi kelemahan utama dari mode ECB, yaitu setiap blok plainteks yang sama akan selalu dienkripsi menjadi blok chiperteks yang sama. Modifikasi ini dilakukan dengan menambahkan satu fungsi baru dengan 2 buah parameter, yaitu nomor blok dan nomor *record* sehingga hasil enkripsi yang diharapkan akan selalu berbeda untuk nomor blok atau nomor *record* yang berbeda. Hasil uji perbandingan menunjukkan bahwa mode ECB yang telah dimodifikasi ini mampu menutupi kelemahan mode ECB seperti yang telah disebutkan di atas.

Kata kunci: enkripsi, dekripsi, blok chiper, ECB, basis data

1. Latar Belakang

Pemrograman web sangat erat kaitannya dengan bidang basis data. Hampir seluruh aplikasi berbasis web yang ada saat ini telah menggunakan basis data. [RAM08] Hal ini terjadi karena sistem basis data telah menjadi komponen yang sangat penting dalam kehidupan manusia sehari-hari. [NUG04] Bahkan basis data telah menjadi suatu kebutuhan di beberapa organisasi dan perusahaan komersial pada saat ini. Basis data digunakan secara luas untuk berbagai bidang seperti bisnis, perbankan, pendidikan, kepegawaian, dan lain-lain. [FAU]

Dengan kebutuhan basis data yang semakin kompleks maka timbul suatu kebutuhan keamanan data dari berbagai macam ancaman diantaranya pembacaan data, modifikasi data dan perusakan data oleh orang yang tidak berhak [SIL02].

Dalam makalah ini, hal yang ingin dihindari adalah pembacaan data yang ada di dalam basis data oleh orang yang tidak berhak. Salah satu caranya adalah dengan mengimplementasikan kriptografi, yaitu dengan mengenkripsi hasil query yang dimasukkan oleh pengguna.

Enkripsi dan dekripsi yang sifatnya acak ini sangat cocok diimplementasikan dengan algoritma *block chipper* mode ECB (*Electronic Code Book*), dengan syarat setiap *record* terdiri dari sejumlah blok diskrit yang sama banyaknya. Mode ECB cocok untuk mengenkripsi file yang diakses secara acak karena tiap blok *plaintext* dienkripsi secara independen. Bahkan jika mode ECB dikerjakan dengan prosesor paralel, maka setiap prosesor dapat melakukan enkripsi atau dekripsi blok plainteks yang berbeda-beda.[MUN06]

ECB yang akan digunakan untuk mengenkripsi atau mendekripsi data adalah ECB yang telah dimodifikasi agar blok chiperteks yang dihasilkan tidak sama meskipun mengenkripsi plainteks yang sama. Hal ini untuk menghindari bagian plainteks yang sering berulang, yang menjadi salah satu kelemahan mode ECB. [MUN06]

2. Rumusan Masalah

Dari latar belakang yang telah diuraikan maka dapat dirumuskan beberapa permasalahan pada makalah ini yaitu :

1. Bagaimana algoritma *block chipper* mode ECB bekerja dalam mengenkripsi atau mendekripsi sembarang data hasil query
2. Bagaimana memodifikasi algoritma *block chipper* mode ECB agar sedemikian rupa tidak menghasilkan blok chiperteks yang sama meskipun mengenkripsi plainteks yang sama untuk blok yang berbeda

3. Tujuan

Dari permasalahan yang ada pada rumusan masalah maka makalah ini bertujuan :

1. Memodifikasi algoritma *block chipper* mode ECB agar aman digunakan sebagai sarana pengamanan data yang merupakan hasil query

4. Batasan Masalah

Makalah menetapkan batasan-batasan masalah sebagai berikut :

1. Makalah ini hanya membahas pemodifikasian algoritma ECB agar chiperteks yang dihasilkan tidak selalu sama untuk plainteks yang sama di blok yang berbeda
2. Data yang akan dienkripsi atau didekripsi adalah data yang ada di dalam basis data yang tiap *record*nya terdiri dari sejumlah blok diskrit yang sama banyaknya.
3. Data yang dienkripsi atau didekripsi adalah data hasil query `SELECT`

5. Dasar Teori

5.1. Algoritma kriptografi modern

Algoritma kriptografi modern umumnya beroperasi dalam mode bit ketimbang mode karakter (seperti yang dilakukan pada chiper subtitusi atau chiper transposisi dari algoritma kriptografi klasik) [MUN] Operasi dalam mode bit berarti semua data dan informasi, baik kunci, plainteks, ataupun chiperteks, dinyatakan dalam rangkaian (string) bit biner, 0 dan 1. Algoritma enkripsi dan dekripsi memroses semua data dan informasi dalam rangkaian bit. Rangkaian bit yang menyatakan plainteks dienkripsi menjadi chiperteks dalam bentuk rangkaian bit, demikian sebaliknya.

5.1.1. Blok chiper

Dalam proses enkripsi atau dekripsi yang memiliki kunci simetri (Algoritma kunci simetri yang merupakan salah satu kategori dari algoritma kriptografi modern [KUR] mengacu pada metode enkripsi yang dalam hal ini baik pengirim maupun penerima memiliki kunci yang sama.), pemrosesan dapat dilakukan dengan dua metode, Cipher aliran (stream cipher) dan Cipher blok (block cipher).

Pada metode cipher blok, proses enkripsi maupun dekripsi dilakukan terhadap sekelompok blok yang terdiri dari sejumlah bit. Panjang bit sudah diketahui sebelumnya dan disesuaikan dengan panjang kunci, biasanya 64 bit atau lebih.

Algoritma enkripsi menghasilkan blok cipherteks yang berukuran sama dengan blok plainteks [LUN]. Dekripsi dilakukan dengan cara yang serupa seperti enkripsi.

Misalkan blok plainteks (P) yang berukuran m bit dinyatakan sebagai vektor

$$P = (p_1, p_2, ..., p_m)$$

yang dalam hal ini p_i adalah bit 0 atau bit 1 untuk i = 1, 2, …, m, dan blok cipherteks (C) adalah

$$C = (c_1, c_2, ..., c_m)$$

yang dalam hal ini c_i adalah bit 0 atau bit 1 untuk i = 1, 2, …, m.

Bila plainteks dibagi menjadi n buah blok, barisan blok-blok plainteks dinyatakan sebagai

$$(P_1, P_2, \ldots, P_n)$$

Untuk setiap blok plainteks P_i, bit-bit penyusunnya dapat dinyatakan sebagai vektor

$$P_i = (p_{i1}, p_{i2}, ..., p_{im})$$

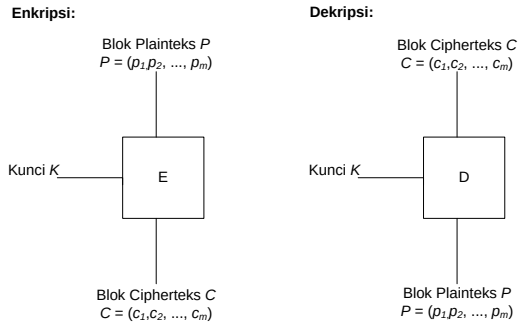
Enkripsi dengan kunci K dinyatakan dengan persamaan

$$E_k(P) = C,$$

sedangkan dekripsi dengan kunci K dinyatakan dengan persamaan

$$D_k(C) = P$$

Proses enkripsi dan dekripsi pada cipher blok diilustrasikan dalam gambar 5.1.



Gambar 5. 1 Enkripsi dan Dekripsi Blok Chiper

Algoritma blok chiper menggabungkan beberapa teknik kriptografi klasik dalam proses enkripsi. Dengan kata lain, chiper blok dapat diacu sebagai super enkripsi. Teknik kripografi klasik yang digunakan adalah:

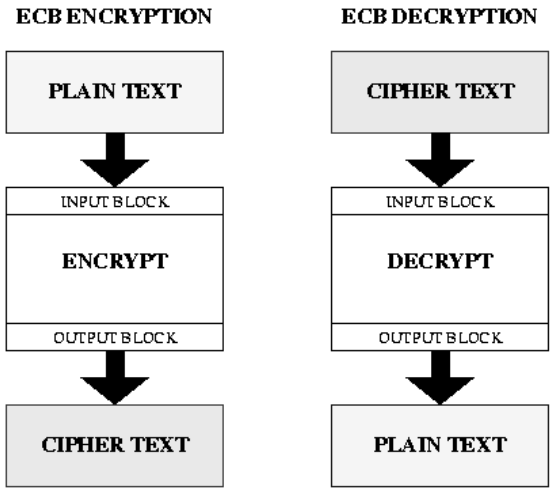
1. Subtitusi
   Teknik ini mengganti satu atau sekumpulan bit pada blok plainteks tanpa mengubah urutannya. Aturan penukaran dapat diperoleh melalui suatu persamaan / fungsi atau menggunakan tabel substitusi (S-box)
2. Transposisi atau permutasi
   Teknik ini memindahkan posisi bit pada blok plainteks berdasarkan aturan tertentu

Selain kedua teknik di atas, chipper blok juga menggunakan dua teknik tambahan, yaitu:

3. Ekspansi
   Teknik ini memperbanyak jumlah bit pada blok plainteks berdasarkan aturan tertentu
4. Kompresi
   Teknik ini kebalikan dari ekspansi, dimana jumlah bit pada blok plainteks diciutkan berdasarkan aturan tertentu.

5.1.2. Mode Electronic Code Book (ECB)

Mode ECB adalah mode yang paling umum dan paling mudah untuk diimplementasikan. Cara yang digunakan adalah dengan membagi data ke dalam blok-blok data terlebih dahulu yang besarnya sudah ditentukan. Blok-blok data inilah yang disebut *plaintext* karena blok data ini belum disandikan. Proses enkripsi akan langsung mengolah *plaintext* menjadi *ciphertext* tanpa melakukan operasi tambahan. Suatu blok *plaintext* yang dienkripsi dengan menggunakan kunci yang sama akan menghasilkan *ciphertext* yang sama.



Gambar 5. 2 Mode Operasi ECB

Pada mode ECB, setiap blok plainteks dienkripsi secara individual dan independen menjadi blok chiperteks. Secara matematis, enkripsi dengan mode ECB dinyatakan sebagai:

$$C_i = E_K(P_i)$$

dan dekripsi sebagai

$$P_i = D_K(C_i)$$

yang dalam hal ini , K adalah kunci dan P_i dan C_i masing-masing blok plainteks dan chiperteks ke-i.

Istilah "*code book*" di dalam ECB muncul dari fakta bahwa blok plainteks yang sama selalu dienkripsi menjadi blok chiperteks yang sama sehingga secara teoritis dimungkinkan membuat buku kode plainteks dan chiperteks yang berkoresponden. Namun semakin besar ukuran blok, semakin besar pula ukuran buku kodenya. Misalkan jika blok berukuran 64 bit, maka buku kode terdiri dari $2^{64} - 1$ buah kode, yang berarti terlalu besar untuk disimpan. Lagipula setiap kunci mempunyai buku kode yang berbeda.

Keuntungan Mode ECB

a. Karena tiap blok plainteks dienkripsi secara independen, maka kita tidak perlu mengkripsi file secara linier. Kita dapat mengekripsi 5 blok pertama, kemudian blok-blok di akhir, dan kembalik ke blok-blok di tengah dan seterusnya. Maka dari itu mode ECB cocok untuk mengenkripsi file yang diakses secara acak, misalnya file basisdata. Jika basisdata dienkripsi dengan mode ECB, maka sembarang record dapat dienkripsi atau didekripsi secara

independen dari record lainnya (dengan asumsi setiap record terdiri dari sejumlah blok diskrit yang sama banyaknya)

b. Jika satu atau lebih bit pada blok chiperteks mengalami kesalahan, maka kesalahan ini hanya mempengaruhi chiperteks yang bersangkutan pada waktu dekripsi. Blok-blok chiperteks yang lainnya bila didekripsi tidak akan terpengaruh oleh kesalahan bit chiperteks tersebut.

Kelemahan Mode ECB

a. Jika ada bagian plainteks yang sering berulang (sehingga terdapat blok-blok plainteks yang sama), maka hasil enkripsinya menghasilkan blok chiperteks yang sama

b. Pihak lawan dapat memanipulasi chiperteks untuk mengelabui penerima pesan, misalnya dengan menghapus atau menyisipkan beberapa buah blok chiperteks.

6. Analisis Masalah

6.1. Masalah

Kebutuhan basis data semakin menjadi kebutuhan yang tidak terpisahkan dari suatu sistem perangkat lunak pada suatu organisasi. Karena itu basis data harus memiliki ketersediaan yang tinggi, yaitu harus dapat diakses kapanpun dan dari manapun. Untuk memenuhi kebutuhan dapat diakses dari manapun maka basis data harus dapat diakses secara remote melalui suatu jaringan komputer. Dengan cara ini pengakses basis data tidak harus berada pada lokasi yang secara fisik terintegrasi dengan lokasi data. Pengakses basis data dapat mengakses basis data selama terdapat jaringan yang menghubungkan lokasi basis data dengan lokasi pengakses dan tersedia antarmuka komunikasi dengan basis data.

Namun pengaksesan data secara remote ini sangat rawan akan pencurian data. Pencurian data ini bisa berupa penyadapan di tengah jalan atau pengaksesan data oleh orang yang tidak terotentifikasi melalui *sql injection*, dan sebagainya.

Salah satu cara untuk mengatasi hal ini adalah dengan mengenkripsi hasil query sehingga data yang dikirimkan akan berupa string yang sudah terenkripsi. String terenkripsi ini kemudian akan didekripsi oleh aplikasi yang ada di sisi pengguna. Algoritma enkripsi yang biasa digunakan adalah dengan menggunakan algoritma blok chiper mode ECB dan juga algoritma RC4. Namun algoritma block chiper mode ECB adalah algoritma yang paling cocok diterapkan untuk pengamanan data seperti ini karena sifatnya yang bisa mengenkripsi data yang diakses secara acak (data sembarang).

Kelemahan algoritma mode ECB adalah jika terdapat plainteks yang sama meskipun dalam blok yang berbeda, maka hasil enkripsinya akan sama, berupa pola-pola chiperteks yang mudah dipecahkan dengan serangan yang berbasis statistik (menggunakan frekuensi kemunculan blok chiperteks).

6.2. Penanganan Masalah

Kelemahan algoritma mode ECB seperti yang telah disebutkan dalam poin 6.1 di atas sebenarnya bisa dikurangi dengan menggunakan ukuran blok yang besar, sebab ukuran blok yang besar dapat menghilangkan kemungkinan menghasilkan blok-blok yang identik.

Namun penggunaan blok yang berukuran besar akan memengaruhi performansi sistem untuk melakukan enkripsi dan dekripsi karena proses enkripsi atau dekripsi melibatkan seluruh bit yang ada dalam 1 blok. Jika 1 blok tersebut berukuran sangat besar, maka jumlah bit yang terkandung dalam blok tersebut juga akan sangat banyak, yang berarti akan memakan waktu dalam komputasi enkripsi atau dekripsi bit-bit dalam blok tersebut.

Cara efektif yang dapat dilakukan adalah dengan memodifikasi algoritma mode ECB sehingga kita tidak perlu menggunakan blok berukuran besar untuk mengurangi kelemahan mode ECB tersebut.

Modifikasi yang dilakukan adalah dengan menambahkan 1 fungsi baru dengan 2 buah parameter, yaitu nomor *record* dan nomor blok. Parameter nomor record dimaksudkan agar chiperteks yang dihasilkan akan selalu berbeda untuk nomor record yang berbeda. Enkripsi terhadap ID = 1 dan ID = 2 pada tabel 6.1 akan menghasilkan chiperteks yang berbeda walaupun plainteksnya sama.

Paramater nomor blok dimaksudkan agar chiperteks yang dihasilkan akan selalu berbeda untuk blok yang berbeda pada nomor *record* yang sama. Misalkan blok plainteks pada tabel 6.1 berukuran 16 bit,

sehingga plainteks untuk ID = 1akan terdiri dari 2 blok kembar, yaitu Aa atau dalam binary, 0100000101100001. Dengan menggunakan fungsi tambahan, enkripsi Aa pada blok pertama tidak akan menghasilkan string chiperteks yang sama dengan enkripsi Aa pada blok kedua.

Tabel 6. 1 Contoh Data

ID	Data
1	AaAa
2	AaAa

Fungsi baru ini akan di XOR kan dengan algoritma mode ECB yang lama sehingga proses enkripsinya akan menjadi:

$$C_i = E_K(P_i) \oplus F(i, r)$$

dan dekripsinya menjadi

$$P_i = D_K(C_i) \oplus F(i, r)$$

dengan i = nomor blok dan r = nomor record.

Fungsi F dalam prakteknya dapat menggunakan perhitungan-perhitungan rumit sehingga kriptanalis akan kesulitan dalam menerka plainteks.

6.3. Perbandingan mode ECB hasil modifikasi

Misalkan

1. Data yang akan dienkripsi adalah data seperti yang ada pada tabel 6.1.
2. Kunci yang digunakan adalah 11 atau dalam binary 0011000100110001.
3. Fungsi enkrips adalah fungsi XOR sederhana
4. Fungsi F menggunakan perhitungan sederhana deret aritmatika 2n + 1 yang dikombinasikan dengan fungsi transposisi

6.3.1. Mode ECB biasa

Jika pengguna memasukkan query SELECT dan sistem mengirimkan hasil berupa *record* 1 yang sudah terenkripsi, maka blok chiperteks yang dihasilkan adalah:

Blok 1: $C_1 = K \oplus P_1$

$= pP\ (0111000001010000)$

Blok 2: $C_2 = K \oplus P_2$

$= pP\ (0111000001010000)$

Sehingga C = pPpP

Jika pengguna memasukkan query SELECT dan sistem mengirimkan hasil berupa *record* 2 yang sudah terenkripsi, maka blok chiperteks yang dihasilkan adalah:

Blok 1: $C_1 = K \oplus P_1$

$= pP\ (0111000001010000)$

Blok 2: $C_2 = K \oplus P_2$

$= pP\ (0111000001010000)$

Sehingga C = pPpP

Dengan menggunakan mode ECB biasa, string AaAa baik pada record 1 maupun record 2 akan menghasilkan chiperteks yang sama, yaitu pPpP.

Terlihat bahwa untuk blok plainteks yang sama, akan dienkripsi menjadi blok chiperteks yang sama. Dan untuk data yang sama, chiperteks yang dihasilkan juga sama meskipun berada dalam nomor record yang berbeda.

Hal ini seharusnya tidak boleh terjadi karena kriptanalis dapat memecahkan plainteks dengan mempelajari kelemahan mode ECB tersebut. Jika suatu waktu kriptanalis mengetahui bahwa blok Aa dienkripsi menjadi blok pP, maka setiap kali ia menemukan chiperteks pP, ia dapat langsung mendekripsinya menjadi Aa.

6.3.2. Mode ECB hasil modifikasi

Hasil perhitungan fungsi F(i,r) setelah menghitung fungsi 2n + 1 , dengan n = nomor blok, yang hasilnya ditransposisikan sebanyak r

F(1,1) = 00000110

F(1,2) = 00001100

F(2,1) = 00001010

F(2,2) = 00010100

Jika pengguna memasukkan query SELECT dan sistem mengirimkan hasil berupa *record* 1 yang sudah terenkripsi, maka blok chiperteks yang dihasilkan adalah:

Blok 1: C_1 = K⊕ P_1⊕ F(1,1)

= pV (0111000001010110)

Blok 2: C_2 = K⊕ P_2⊕ F(1,2)

= p\ (0111000001011100)

Sehingga C = pVp\

Jika pengguna memasukkan query SELECT dan sistem mengirimkan hasil berupa *record* 2 yang sudah terenkripsi, maka blok chiperteks yang dihasilkan adalah:

Blok 1: C_1 = K⊕ P_1⊕ F(2,1)

= pZ (0111000001011010)

Blok 2: C_2 = K⊕ P_2⊕ F(2,2)

= pD (0111000001000100)

Sehingga C = pZpD

Dengan menggunakan mode ECB yang sudah dimodifikasi, string AaAa akan dienkripsi menjadi string yang berbeda untuk nomor record yang berbeda. Bahkan blok Aa pun dienkripsi menjadi string yang berbeda untuk nomor blok yang berbeda meskipun masih dalam nomor record yang sama.

Mode ECB yang sudah dimodifikasi ini terbukti bisa mengatasi kelemahan mode ECB seperti yang telah disebutkan sebelumnya. Jika suatu waktu kriptanalis mengetahui bahwa suatu blok Aa dienkripsi menjadi blok pV, maka ketika ia menemukan chiperteks pV, ia tidak dapat langsung mendekripsinya menjadi Aa karena blok pV belum tentu hasil dekripsinya adalah blok Aa.

7. Kesimpulan

Kesimpulan yang dapat diambil pada studi dan modifikasi algoritma ECB ini adalah:

1. Algoritma block chiper mode operasi ECB adalah algoritma yang cocok diterapkan dalam pengamanan data di basis data karena sifatnya dalam mengenkripsi atau mendekripsi tiap blok plainteks secara independen sehingga sangat cocok untuk mengenkripsi atau mendekripsi file yang diakses secara acak
2. Algoritma block chiper mode operasi ECB biasa belum cukup memadai untuk pengamanan data karena memiliki beberapa kelemahan, diantaranya blok plainteks yang sama akan selalu menghasilkan blok chiperteks yang sama
3. Untuk mengatasi kelemahan tersebut, maka diperlukan sebuah fungsi tambahan dengan 2 buah parameter sehingga akan dihasilkan blok chiperteks yang berbeda untuk blok plainteks yang sama namun berbeda nomor blok atau nomor *record*.

8. Daftar Pustaka

[RAM08] Ramadhan, Jimmy Kharisma, 2008, *Pembangunan Form Generator Berbasis Web untuk Membangkitkan Form Input dari Basis Data*.

[NUG04] Nugroho, Adi, 2004, *Konsep Pengembangan Sistem Basis Data*, Informatika: Bandung

[FAU] Fauzan, Mohamad Firda, *Pengamanan Transmisi Hasil dan Data Query Basis Data dengan Algoritma Kriptografi RC4*

[SIL02] A. Silberschatz, H. F.Korth, S.Sudarshan, 1999, *Database System Concepts, 4rd edition*, McGraw-Hill

[MUN06] Munir, Rinaldi, 2006, *Kriptografi*, Informatika: Bandung

[KUR04] Kurniawan, Yusuf, 2004, *Kriptografi, Keamanan Internet dan Jaringan Komunikasi*, Informatika: Bandung

[LUN] Lung, Chan, Studi dan Implementasi Advanced Encryption Standard dengan Empat Mode Operasi Block Cipher