Studi Algoritma
Cellular Message Encryption Algorithm (CMEA)

Rian Permata Putra - 13504083
Program Studi Teknik Informatika
Institut Teknologi Bandung
Jl Ganesha 10 Bandung 40132
Email : if14083@students.if.itb.ac.id

Abstract – *CMEA sudah digunakan secara luas untuk keamanan wireless. Pada saat ini cara untuk memecahkan CMEA telah ditemukan. Untuk itu dibutuhkan sebuah algoritma yang lebih baik daripada CMEA. Makalah ini akan membahas alasan mengapa CMEA bisa sangat lemah. Modifikasi yang akan dilakukan terhadap CMEA yang bernama CMEA-I, juga analisis terhadapa keamanan yang ditawarkan CMEA-I*

Kata Kunci *: CMEA, crytanalysis,wireless secutiry*

1. PENDAHULUAN

Dalam kriptografi, Cellular Message Encryption Algorithm (CMEA) adalah block cipher yang digunakan untuk mengenkrpsi data digital telepon seluler. CMEA adalah salah satu dari empat primitif kryptografik dikembangkan oleh Telecommunications Industry Association (TIA) yang dikhususkan untuk telekomunikasi dan didesign untuk enkripsi control channel daripada data suara. CMEA menggunakan kunci 64 bit.

Pada Maret 1997, Counterpane System dan UC Berkeley bekerja sama mempublikasikan serangan pada cipher yang memperlihatkan bahwa CMEA mempunyai beberapa kelemahan. Serangan ini menggunakan 40-80 known plainteks, mempunyai waktu complexity 2^{24}-2^{32} dan selesai hanya dalam hitungan menit atau jam pada standar workstation.

2. DESKRIPSI CMEA

CMEA adalah block cipher yang berorientasi byte dengan kunci 64 bit. Ukuran blok bisa berapa saja dalam byte. CMEA cukup sederhana, dan optimal untuk microprocessor 8-bit dengan sumber yang terbatas.

CMEA terdiri dari tiga lapisan. Lapisan pertama melakukan satu pass non-linear pada blok. Ini akan berpengaruh pada penyebaran kiri ke kanan. Lapisan kedua murni linear, bertujuan untuk membuat. Langkah kedua melakukan operasi XOR pada setengah bagian kanan blok ke setengah bagian kiri blok. Lapisan ketiga melakukan pass non-linear terakhir pada blok dari kanan ke kiri, invers dari lapisan pertama.

CMEA memperoleh non-linear pada lapisan pertama dan ketiga dari tabel lookup kunci 8-bit yang dikenal sebagai T-box. T-box menghitung keluaran 8-bit sebagai :

$$T(x) = C(((C(((C(((C((x \oplus K0) + K1) + x) \oplus K2) + K3) + x) \oplus K4) + K5) + x) \oplus K6) + K7) + x$$

Diberikan input x dan kunci 8-byte $K_{0..7}$. Pada persamaan ini C adalah tabel unkeyed 8-bit lookup yang dineal sebagai CaveTable. CaveTable diberikan pada gambar 1.

Hi\lo	.0	.1	.2	.3	.4	.5	.6	.7	.8	.9	.a	.b	.c	.d	.e	.f
0.	d9	23	5f	e6	ca	68	97	b0	7b	f2	0c	34	11	a5	8d	4e
1.	0a	46	77	8d	10	9f	5e	62	f1	34	ec	a5	c9	b3	d8	2b
2.	59	47	e3	d2	ff	ae	64	ca	15	8b	7d	38	21	bc	96	00
3.	49	56	23	15	97	e4	cb	6f	f2	70	3c	88	ba	d1	0d	ae
4.	e2	38	ba	44	9f	83	5d	1c	de	ab	c7	65	f1	76	09	20
5.	86	bd	0a	f1	3c	a7	29	93	cb	45	5f	e8	10	74	62	de
6.	b8	77	80	d1	12	26	ac	6d	e9	cf	f3	54	3a	0b	95	4e
7.	b1	30	a4	96	f8	57	49	8e	05	1f	62	7c	c3	2b	da	ed
8.	bb	86	0d	7a	97	13	6c	4e	51	30	e5	f2	2f	d8	c4	a9
9.	91	76	f0	17	43	38	29	84	a2	db	ef	65	5e	ca	0d	bc
a .	e7	fa	d8	81	6f	00	14	42	25	7c	5d	c9	9e	b6	33	ab
b.	5a	6f	9b	d9	fe	71	44	c5	37	a2	88	2d	00	b6	13	ec
c.	4e	96	a8	5a	b5	d7	c3	8d	3f	f2	ec	04	60	71	1b	29
d.	04	79	e3	c7	1b	66	81	4a	25	9d	dc	5f	3e	b0	f8	a2
e.	91	34	f6	5c	67	89	73	05	22	aa	cb	ee	bf	18	d0	4d
f.	f5	36	ae	01	2f	94	c3	49	8b	bd	58	12	e0	77	6c	da

Gambar 1. CaveTable

Algoritma melakukan enkripsi pesan *n*-byte $P_{0,...,n-1}$ ke ciphertext $C_{0,...,n-1}$ dengan kunci $K_{0...7}$ sebagai berikut :

$y_0 \leftarrow 0$
for $i \leftarrow 0;...;n-1$
  $P'_i \leftarrow P_i + T(y_i \oplus i)$
  $y_{i+1} \leftarrow y_i + P'_i$

for $i \leftarrow 0;...;n/2 - 1$
  $P''_i \leftarrow P'_i \oplus (P'_{n-1-i} \square 1)$
$Z0 \leftarrow 0$
for $i \leftarrow;...; n - 1$
  $z_{i+1} \leftarrow z_i + P''_i$
  $C_i \leftarrow P''_i - T(z_i \oplus i)$

Keterangan :

+ dan - : penambahan dan pengurangan modulo 256
⊕ : logical bitwise ekslusif OR (XOR)
 : logical bitwise OR.
T : fungsi yang didefinisikan sebelumnya.

3. SERANGAN PADA CMEA

Serangan pada CMEA dapat dibagi dua tipe yaitu :

3.1. Serangan Chosen Plaintext

Serangan chosen plaintext adalah serangan dengan cara cryptanalyst mampu untuk mendefinisikan sendiri plaintext, dimasukkan kecipher dan kemudian melakukan analisa pada ciphertext.

CMEA sangat lemah terhadap serangan ini. Seseorang dapat memperoleh semua T-box dengan 338 teks terpilih dan sedikit sekali usaha. Penyerang tidak harus mengetahui panjang blok kunci. Penyerang akan melakukan langkah berikut :

1) Mendapatkan T (0)
   Untuk setiap nilai x, dimana x adalah byte, pesan P = (1−x, 1−x, 1−x, · · · , 1−x) terenkripsi, dimana tanda – menunjukkan pengurangan dalam binary. Tiap byte mempunyai nilai (1 – x). Jika hasil cipher C = (-x, ...) maka kemungkinan besar T (0) = x. Hanya ada 256-92 = 164 kemungkinan nilai T(0), sehingga nilai benar diharapkan bisa didapatkan dalam rata-rata 164/2 = 82 percobaan.

2) Mendapatkan nilai T-box yang tersisa
   Untuk setiap byte j, untuk mendapatkan nilai T(j), misalkan k = ((n − 1) ⊕ j) − (n − 2), dengan blok yang di inginkan sepanjang n bytes. Enkripsi dari P = (1 − T (0), 1 − T (0), · · · , 1 − T (0), k − T(0), 0). Jika hasil dari C = (t−T (0), · · ·) maka kemungkinan besar T(j) = t, dengan kemungkinan ambigius dalam LSB. Tahap kedua membutuhkan 256 plainteks terpilih, sehingga total membutuhkan 256 plainteks terpilih.

4. MODIFIKASI CMEA

4.1. CMEA-I

Ada tiga modifikasi yang bisa dilakukan yaitu :

- Modifikasi 1 : update persamaan dari P_i
  Persamaan yang baru berbentuk :

$$P'_i = P_i + T\ (y_i \oplus f(i, n))$$

Fungsi f(i,n) harus sedemikian rupa sehingga T-box bisa diakses dari beberapa titik berbeda. Setelah mempertimbangkan berbagai bentuk, maka fungsi yang dipilih adalah f(i,n) = (2i)%n, % adalah operasi modulo.
Sehingga persamaan menjadi

$$P'i = Pi + T\ (yi \oplus (2i)\%n)$$

Dan algoritmanya menjadi

$y_0 = 0$

for (i = 0; i < n; i + +)
{
$P'_i = P_i + T\ (y_{i \oplus} ((2i)\%n))$
$y_{i+1} = y_i + P'i$
}

For (i = 0; i < bn/2c; i + +)
$P''_i = P'_i \oplus (P'_{n-i-1}$ 1)

$z_0 = 0$

for(i = 0; i < n; i + +)
{
$z_i + 1 = z_i + P''_i$
$C_i = P''_i - T\ (z_i \oplus ((2i)\%n))$
}

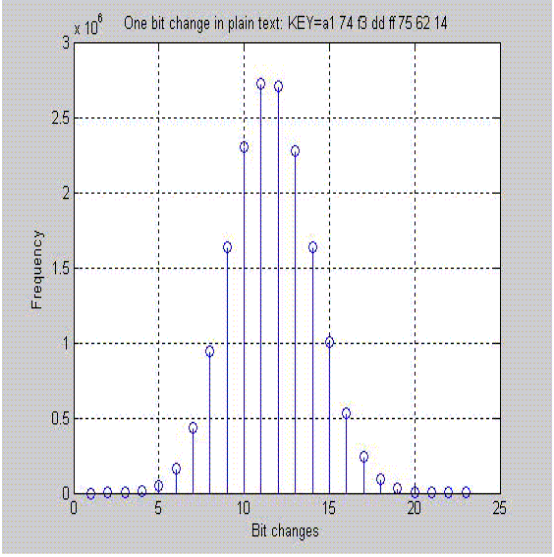
- Modifikasi 2: Tabel Cave diganti dengan AES S-box.
- Modifikasi 3: T-box yang sebelumnya hanya mempunyai 4 round, dengan meningkatkan menjadi 8 round bisa mencegah serangan meet-in-the-middle.

4.1 Analisis Keamanan CMEA-I

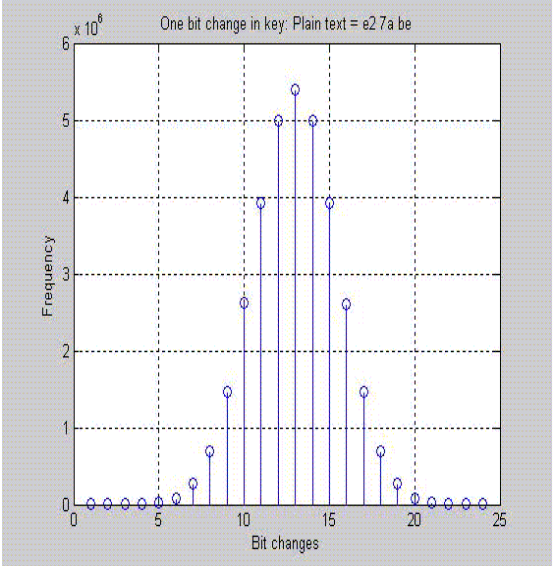
4.1.1 Diffusion dan Confusion pada Algoritma CMEA-I

Diffusion dan confusion adalah dua properti penting dalam keamanan block ciphers. Kriteria diffusion mewajibkan bahwa perubahan pada satu bit dari plaintext harus menyebabkan beberapa perubahan pada ciphertext. Untuk menguji properti diffusion pada CMEA-I, algoritma diterapkan pada sepasang plaintext yang berbeda satu bit.Jumlah keluaran bit yang terpengaruh harus mempunyai rata-rata n/2, dimana n adalah jumlah bits dalam cipher. Dengan kata lain diharapkan untuk cipher yang baik kurang lebih setengah dari bits keluaran harus terkena pengaruh.Percobaan dilakuakn pada ukuran blok sebesar tiga bytes (24 bits).). Pada gambar 1, frequensi jumlah bits yang terpengaruh dibandingkan dengan jumlah bits yang terpengaruh. Perbandingan menunjukkan kira-kira 12 bits terpengaruh untuk jumlah maksimal kasus. Properti diffusion dipenuhi oleh CMEA-I.
Confusion mewajibkan bahwa perubahan pada satu bit pada kunci harus menyebabkan perubahan pada

beberapa bits pada ciphertext. Untuk menguji properti confusion pada algoritma CMEA-I digunakan sepasang kunci yang berbeda satu bit. Jumlah keluaran yang terpengaruh menurut kriteria Avalanche harus berkisar n/2, dimana n adalah jumlah bits dalam cipher. Percobaan dilakukan pada ukuran blok tiga bytes (24 bits). Pada gambar 2, frequensi jumlah bits yang terpengaruh dibandingkan dengan jumlah bits yang terpengaruh. Perbandingan menunjukkan kira-kira 12 bits terpengaruh untuk jumlah maksimal kasus. Properti confusion dipenuhi oleh CMEA-I.



Gambar 1: Efek Avalanche effect untuk menunjukkan diffusion

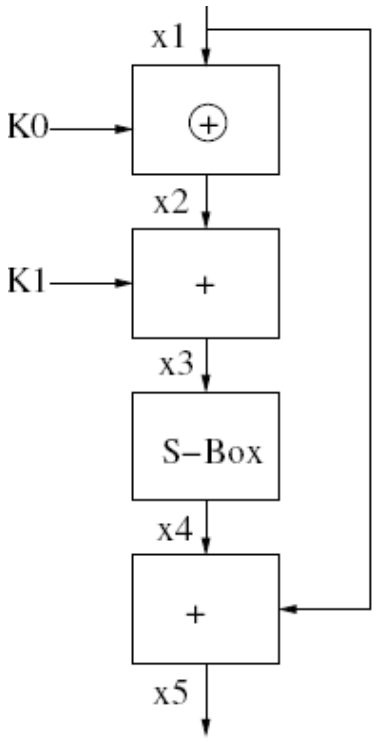


Gambar 2: Efek Avalanche effect untuk menunjukkan confusion

4.1.2 Differemtial Analisis Pada T-Box

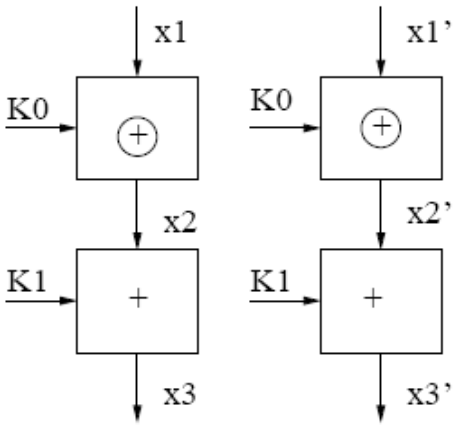
Differntial analisis pada cipher block meninjau bahwa jika diberikan input yang berbeda maka besar kemungkianan akan dihasilkan output yang berbeda. Pada ideal cipher untuk n-bit block maka kemungkinannya sebesar $1/2^n$. Differential cryptanalisis akan mencari kemungkinan keluaran yang berbeda akan muncul jika diberikan input yang berbeda.

Gamabr 3 menunjukkan keamanan menggunakan satu roung T-Box.



Gambar 3. One roung T-Box

Diberikan x1 dan x5, seseorang bisa meghitung x3. Demikian mengurangi masalah cryptoanalisis membagi T-Box seperti pada gambar 4.



Gambar 4. Diffential Analsis dari T-Box

Disini terlihat bahwa satu round T-box memiliki kelemahan. Oleh sebab itu untuk mengatasi masalah tersebut adalah dengan menambah round pada T-box, dalah makalah ini sejumlah 8 round.

5. KESIMPULAN

Pada makalah ini telah dilakukan modifikasi algoritma CMEA menjadi CMEA-I. Makalah ini menunjukkan bahwa cryptanalysis sekarang gagal untuk memecahkan CMEA-I. Dengan modifikasi, algoritma CMEA bisa dibuat lebih kuat dan sebab itu sesuai digunakan untuk jaringan keamanan wireless.

DAFTAR REFERENSI

[1] Debdeep Mukhopadhyay , Dipanwita Roy Chowdhury, Customizing Cellular Message Encryption Algorithm, Department of Computer Science and Engineering, IIT Kharagpur.

[2] Greg Rose, Authentication and Security in Mobile Phones, QUALCOMM Australiaz

[3] David Wagner, Bruce Schneier, John Kelsey, Cryptanalysis of the Cellular Message Encryption Algorithm, University of California, Berkeley Counterpane Systems