ANALISIS SERANGAN KRIPTANALISIS LINEAR MATSUI PADA DES

Ramon Rusli – NIM 13506025

Program Studi Teknik Informatika Institut Teknologi Bandung
Jalan Ganesha 10, Bandung
Email: if16025@students.if.itb.ac.id

Abstrak

Makalah ini membahas tentang studi analisis serangan kriptanalis linear yang diterapkan oleh Mitsuru Matsui terhadap *block cipher*. *Data Encryption Standard*, yang biasa disingkat DES. DES adalah algoritma *block cipher* yang populer karena menjadi standard algoritma enkripsi kunci simetri, dan penilaian kekuatannya telah diuji oleh Nasional Security Agency (NSA) Amerika Serikat dan telah disetujui oleh National Bureau of Standard (NBS). DES merupakan *block cipher* yang beroperasi dengan blok berukuran 64-bit dan kunci 56-bit. DES telah diimplementasikan dalam perangkat keras dan juga perangkat lunak, di mana keduanya didesain dalam sistem komputer maupun jaringan computer.

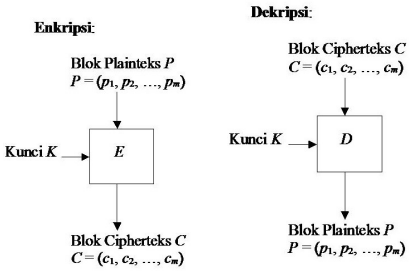
Namun saat ini standard ini telah tergantikan oleh algoritma-algoritma lain karena DES dianggap tidak aman lagi. Pada saat ini, DES telah mengalami banyak perkembangan karenanya, sehingga muncul algoritma-algoritma enkripsi kunci simetri lainnya, seperti AES, Triple DES, dan DES-X. Hal ini disebabkan karena adanya serangan-serangan yang dilakukan terhadap algoritma ini, yang dulunya diyakini memiliki tingkat keamanan yang sangat tinggi. Eksperiman serangan pertama yang diketahui adalah serangan kriptanalisis linear yang ditemukan oleh Mitsuru Matsui yang membutuhkan 2^{43} *known plaintext*, dan diketahui sebagai serangan terkuat untuk algoritma DES. Metode ini telah diimplementasikan dan merupakan eksperimen terhadap DES yang pertama yang dilaporkan. Tidak ada bukti bahwa DES kebal terhadap serangan ini. Kriptanalisis linear telah menjadi salah satu teknik umum yang dapat diterapkan ke banyak skema algoritma.

***Kata kunci**: Data Encryption Standard*, *block cipher*, kriptanalisis linear, *known plaintext*

1. Pendahuluan

1.1. Block cipher

Kriptografi Block Chiper bekerja pada suatu data yang berbentuk blok/kelompok data dengan panjang data tertentu (dalam beberapa byte), yang berarti dalam sekali proses enkripsi atau dekripsi data yang masuk mempunyai ukuran yang sama. Panjang kunci yang digunakan dalam melakukan enkripsi dan dekripsi terhadap *block cipher* adalah sama dengan panjang block.



Pada algoritma *block cipher*, plainteks yang masuk akan diproses dengan panjang blok yang tetap yaitu n, namun terkadang jika ukuran data ini terlalu panjang maka dilakukan pemecahan dalam bentuk blok yang lebih kecil. Jika dalam pemecahan dihasilkan blok data yang kurang dari jumlah data dalam blok maka akan dilakukan proses *padding* (penambahan beberapa bit).

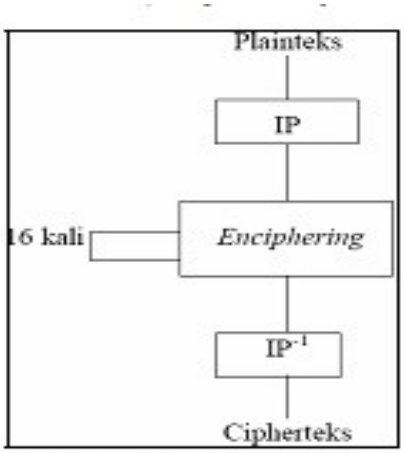
1.2. DES (Data Encryption Standard)

1.2.1 Sejarah DES

Algoritma DES dikembangkan di IBM dibawah kepemimpinan W. L. Tuchman pada tahun 1972. Pada tahun 1976, dilakukan workshop pertama terhadap DES, dan pada tahun yang sama, DES dipilih sebagai Federal Information Processing Standard (FIPS) untuk negara Amerika Serikat, Algoritma DES telah disetujui oleh National Bureau of Standard (NBS) setelah penilaian kekuatannya oleh National Security Agency (NSA) Amerika Serikat. Algoritma ini didasarkan pada algoritma Lucifer yang dibuat oleh Horst Feistel.

1.2.2 Skema DES

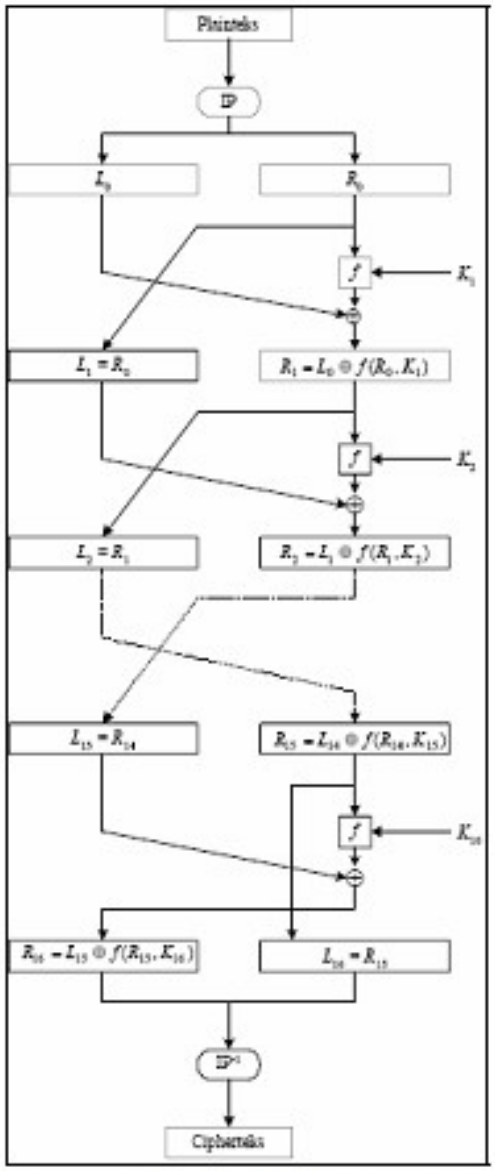
DES termasuk ke dalam sistem kriptografi simetri dan tergolong jenis cipher blok. DES beroperasi pada ukuran blok 64 bit. DES mengenkripsikan 64 bit plainteks menjadi 64 bit cipherteks dengan menggunakan 56 bit kunci internal (internal key) atau subkey. Kunci internal dibangkitkan dari kunci eksternal (external key) yang panjangnya 64 bit.



Skema global dari algoritma DES adalah sebagai berikut:

1. Blok plainteks dipermutasi dengan matriks permutasi awal (initial permutation atau IP).
2. Pada hasil permutasi awal kemudian dilakukan proses enciphering sebanyak 16 kali (16 putaran). Setiap putaran menggunakan kunci internal yang berbeda.
3. Hasil proses enciphering kemudian dipermutasi dengan matriks permutasi balikan (invers initial permutation atau IP^{-1}) menjadi blok cipherteks.

Di dalam proses enchipering, blok plainteks terbagi menjadi dua bagian, kiri (L) dan kanan (R), yang masing- masing panjangnya 32 bit. Kedua bagian ini masuk ke dalam 16 putaran DES. Pada setiap putaran i, blok R merupakan masukan untuk fungsi transformasi yang disebut f. Pada fungsi f, blok R dikombinasikan dengan kunci internal Ki. Keluaran dari fungsi f di-XOR- kan dengan blok L untuk mendapatkan blok R yang baru. Sedangkan blok L yang baru langsung diambil dari blok R sebelumnya. Ini adalah satu putaran DES. Satu putaran DES merupakan model jaringan Feistel. Pada gambar di bawah ini, diperlihatkan skema enkripsi pada algoritma DES yang lebih rinci.



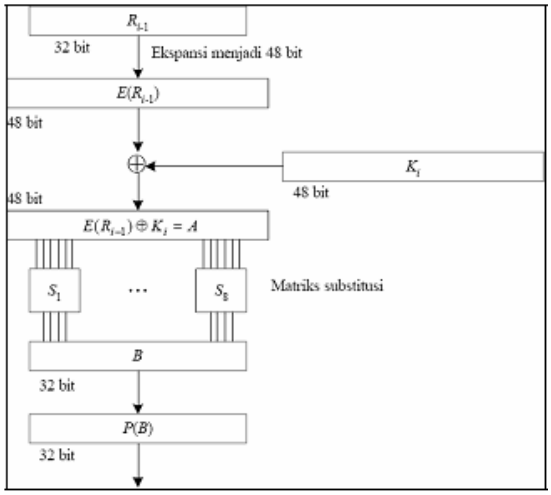
Perlu dicatat dari gambar di atas bahwa jika (L_{16}, R_{16}) merupakan keluaran dari putaran ke-16,

maka (R_{16}, L_{16}) merupakan pra- cipherteks (pre-ciphertext) dari enciphering ini. Cipherteks yang sebenarnya diperoleh dengan melakukan permutasi awal balikan, IP^{-1}, terhadap blok pra-cipherteks.

Proses enkripsi terhadap blok plainteks dilakukan setelah permutasi awal. tiap blok plainteks mengalami 16 kali putaran enciphering). Setiap putaran enciphering merupakan jaringan Feistel yang secara matematis dinyatakan sebagai

$$L_i = R_{i-1}$$
$$R_i = L_{i-1} \oplus f(R_{i-1}, K_i)$$

Diagram komputasi fungsi f diperlihatkan pada gambar di bawah ini:



Pada gambar di atas, E adalah fungsi ekspansi yang memperluas blok Ri – 1 yang panjangnya 32- bit menjadi blok 48 bit. Selanjutnya, hasil ekpansi, yaitu E(Ri – 1), yang panjangnya 48 bit di- XOR- kan dengan Ki yang panjangnya 48 bit menghasilkan vektor A yang panjangnya 48- bit:

$$E(R_{i-1}) \oplus K_i = A$$

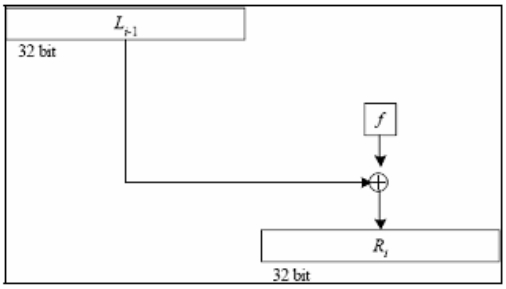
Vektor A dikelompokkan menjadi 8 kelompok, masing- masing 6 bit, dan menjadi masukan bagi proses substitusi. Proses substitusi dilakukan dengan menggunakan delapan buah kotak- S (S-box), S1 sampai S8. Setiap kotak- S menerima masukan 6 bit dan menghasilkan keluaran 4 bit. Kelompok 6- bit pertama menggunakan S1, kelompok 6- bit kedua menggunakan S2, dan seterusnya.

Keluaran proses substitusi adalah vektor B yang panjangnya 48 bit. Vektor B menjadi masukan untuk proses permutasi. Tujuan permutasi adalah untuk mengacak hasil proses substitusi kotak- S. Permutasi dilakukan dengan menggunakan matriks permutasi P (P- box) sbb:
Bit- bit P(B) merupakan keluaran dari fungsi f. Akhirnya, bit- bit P(B) di- XOR- kan dengan Li – 1 untuk mendapatkan Ri

$$Ri = Li - 1 \oplus P(B)$$

Jadi, keluaran dari putaran ke- i adalah

(Li, Ri) = (Ri – 1 , Li – 1 ⊕ P(B))



Proses dekripsi terhadap cipherteks merupakan kebalikan dari proses enkripsi. DES menggunakan algoritma yang sama untuk proses enkripsi dan dekripsi. Jika pada proses enkripsi urutan kunci internal yang digunakan adalah K_1, K_2, …, K_{16}, maka pada proses dekripsi urutan kunci yang digunakan adalah K_{16}, K_{15}, …, K_1. Untuk tiap putaran 16, 15, …, 1, keluaran pada setiap putaran deciphering adalah

$$L_i = R_{i-1}$$
$$R_i = L_{i-1} \oplus f(R_{i-1}, K_i)$$

yang dalam hal ini, (R_{16}, L_{16}) adalah blok masukan awal untuk deciphering. Blok (R_{16}, L_{16}) diperoleh dengan mempermutasikan cipherteks dengan matriks permutasi IP^{-1}. Pra- keluaran dari deciphering adalah (L_0, R_0). Dengan permutasi awal IP akan didapatkan kembali blok plainteks semula.

DES mempunyai beberapa kunci lemah (weak key). Kunci lemah menyebabkan kunci- kunci internal pada setiap putaran sama ($K_1 = K_2 = \ldots = K_{16}$). Akibatnya, enkripsi dua kali berturut-turut terhadap plainteks menghasilkan kembali

plainteks semula.Selain kunci lemah, DES juga mempunyai sejumlah pasangan kunci setengah-lemah (semiweak key). Pasangan kunci setengah- lemah mengenkripsikan plainteks menjadi cipherteks yang sama. Sehingga, satu kunci dalam pasangan itu dapat mendekripsi pesan yang dienkripsi oleh kunci yang lain di dalam pasangan itu.

1.2.3 Keamanan DES

Isu- isu yang menjadi perdebatan kontroversial menyangkut keamanan DES:
1. Panjang kunci
Panjang kunci eksternal DES hanya 64 bit atau 8 karakter, itupun yang dipakai hanya 56 bit. Pada rancangan awal, panjang kunci yang diusulkan IBM adalah 128 bit, tetapi atas permintaan NSA, panjang kunci diperkecil menjadi 56 bit. Alasan pengurangan tidak diumumkan. Tetapi, dengan panjang kunci 56 bit akan terdapat 256 atau 72.057.594.037.927.936 kemungkinan kunci. Jika diasumsikan serangan exhaustive key search dengan menggunakan prosesor paralel mencoba setengah dari jumlah kemungkinan kunci itu, maka dalam satu detik dapat dikerjakan satu juta serangan. Jadi seluruhnya diperlukan 1142 tahun untuk menemukan kunci yang benar.

Tahun 1998, Electronic Frontier Foundation (EFF) merancang dan membuat perangkat keras khusus untuk menemukan kunci DES secara exhaustive search key dengan biaya $250.000 dan diharapkan dapat menemukan kunci selama 5 hari.

Tahun 1999, kombinasi perangkat keras EFF dengan kolaborasi internet yang melibatkan lebih dari 100.000 komputer dapat menemukan kunci DES kurang dari 1 hari.

2. Jumlah putaran
Sebenarnya, delapan putaran sudah cukup untuk membuat cipherteks sebagai fungsi acak dari setiap bit plainteks dan setiap bit cipherteks. Jadi, mengapa harus 16 kali putaran? Dari penelitian, DES dengan jumlah putaran yang kurang dari 16 ternyata dapat dipecahkan dengan known-plaintext attack lebih mangkus daripada dengan brute force attack.

3. Kotak- S
Pengisian kotak- S DES masih menjadi misteri tanpa ada alasan mengapa memilih konstanta-konstanta di dalam kotak itu.

1.2. Serangan terhadap DES

Serangan terhadap DES dibagi menjadi 2 bagian, yaitu serangan brute force dan serangan yang lebih cepat dari brute force, yang biasa diistilahkan sebagai certificational weaknesses. Namun dalam makalah ini, serangan brute force tidak dibahas lebih lanjut. Ada tiga serangan yang diketahui dapat memecahkan enam belas putaran DES dengan kompleksitas lebih rendah daripada algoritma brute force. Namun serangan ini hanya sebatas teori dan tidak memungkinkan untuk dilaksanakan. Ketiga serangan itu adalah:

1. Kriptanalisis differential
2. Kriptanalisis linear
3. Serangan Davies

Dalam makalah ini, pembahasan serangan terhadap DES hanya dilakukan pada serangan kriptanalisis linear saja, tepatnya serangan kriptanalis linear yang ditemukan oleh Mitsuru Matsui.

2. Pembahasan

2.1. Kriptanalisis Linear

Eksperimen serangan pertama terhadap DES yang diketahui adalah serangan kriptanalisis linear yang ditemukan oleh Mitsuru Matsui yang membutuhkan 2^{43} known plaintext, dan diketahui sebagai serangan terkuat untuk algoritma DES. Setelahnya, serangan kriptanalisis tersebut dikenal sebagai Serangan Matsui. Metode ini telah diimplementasikan dan merupakan eksperimen terhadap DES yang pertama yang dilaporkan. Tidak ada bukti bahwa DES kebal terhadap serangan ini. Kriptanalisis linear telah menjadi salah satu teknik umum yang dapat diterapkan ke banyak skema algoritma.

Kriptanalisis linear mencoba untuk memanfaatkan keuntungan dari tingginya tingkat kemunculan ekspresi linear yang melibatkan bit dari plaintext, chipertext, dan juga subkey. Ide dasarnya adalah memperkirakan operasi dari sebagian dari chipertext yang memiliki ekspresi yang linear dengan operasi XOR. Untuk itu dibuatlah langkah linear statistik antara input dan output bits dalam S-box pada algoritma DES. Pada dasarnya, akan terdapat banyak perkiraan ekspresi linear untuk sebuat algoritma chiper,

dan tujuan dari kriptanalisis linear adalah untuk menemukan ekspresi linear yang efektif.

2.2. **Skema Serangan Matsui**

Inti dari serangan Matsui adalah ekspresi linear yang tidak seimbang/unbalance, seperti XOR Sebuah ekspresi linear disebut unbalance bila hasil ekspresi bernilai 1 untuk p=1/2 + e, 0<|e| <1/2 jika plainteks dan kunci independent dan dipilih secara random. Jika diberikan bit plainteks $P_{[i1,...Pir]}$, bit ciphertext $C_{[j1,...js]}$, dan bit kunci K[k1…kt], dengan menggunakan notasi $X_{l1}\oplus... \oplus X_{lu} = X_{[l1...lu]}$, maka ekspresi linear L dapat ditulis dengan:

$$\mathcal{L} : \mathsf{P}_{[i_1,\ldots,i_r]} \oplus \mathsf{C}_{[j_1,\ldots,j_s]} = \mathsf{K}_{[k_1,\ldots,k_t]}$$

Kriptanalis pada serangan Matsui beroperasi pada 14 putaran menggunakan 2 ekspresi bias linear yang mendapatkan informasi statistik pada 26 bit yang berasal dari awal dan akhir putaran subkey. Ke-30 bit kunci sisanya harus dicari dengan teknik exhaustive search. Ekspresi linear L dengan menggunakan dua kondisi fungsi F inidapat dituliskan dengan:

$$\mathcal{L} : \quad \mathsf{P}_{[i_1,\ldots,i_r]} \oplus \mathsf{C}_{[j_1,\ldots,j_s]} \oplus F^{(1)}_{[l_1,\ldots,l_u]}\left(\mathsf{P}, \mathsf{K}^{(1)}\right) \oplus F^{(16)}_{[m_1,\ldots,m_v]}\left(\mathsf{C}, \mathsf{K}^{(16)}\right) = \mathsf{K}_{[k_1,\ldots,k_t]}$$

$F^{(1)}_{[l1..lu]}$ $(P,K^{(1)})$ adalah hasil XOR antara output fungsi F pada putaran pertama dan subkey dari putaran pertama. Notasi yang sama berlaku untuk fungsi F kedua. Ide utama dari serangan ini menggunakan asumsi bahwa untuk setiap ekspresi L yang beroperasi dalam n putaran, di mana kemungkinan

$$\left|\Pr\left[\mathcal{L} = 0 | \mathsf{K}^{(1)} = k^{(1)}, \ldots, \mathsf{K}^{(n)} = k^{(n)}\right] - \frac{1}{2}\right|$$

Adalah besar untuk semua nilai $k^{(1)}$, $k^{(2)}$,…$k^{(n)}$, maka pernyataan berikut bernilai benar:

$$\tau = \frac{\left|\Pr\left[\mathcal{L} = 0 | \mathsf{K} = k_r\right] - \frac{1}{2}\right|}{\left|\Pr\left[\mathcal{L} = 0 | \mathsf{K} = \hat{k}\right] - \frac{1}{2}\right|} \gg 1 \quad \forall \hat{k} \neq k_r$$

Di mana k_r adalah kunci yang benar

Proses dekripsi terhadap putaran pertama dan terakhir dengan menggunakan kandidat subkey yang salah dapat diartikan sebagai dua putaran enkripsi. Dengan begitu, plaintext dan ciphertext akan semakin tidak berhubungan dan ekspresi linear akan semakin bias.

Kriptanalisis linear fase pertama berisi proses mengevaluasi kebiasan dari kedua ekspresi linear untuk semua kandidat subkey yang mungkin dan untuk semua pasangan all-known plaintext-ciphertext.

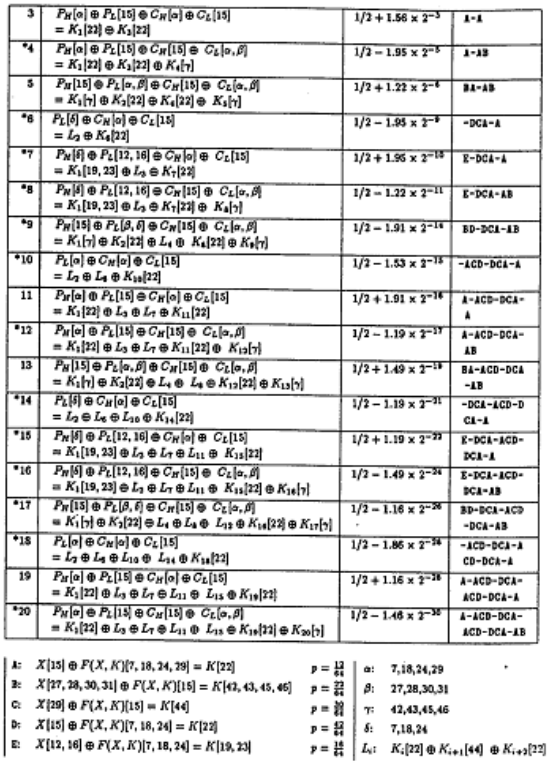
1: N = number of known plaintext-ciphertext pairs at disposal.
2: **for** linear expressions $\mathcal{L}_1$ and $\mathcal{L}_2$ **do**
3: **for all** subkey candidates $\hat{k}_i, 1 \leq i \leq 2^{12}$ **do**
4: $C_{\hat{k}_i}$ = number of times out of N where left part of (2) is equal to 0 when $\mathsf{K} = \hat{k}_i$.
5: **end for**
6: **end for**

Pada fase kedua, kedua list kandidat subkey yang merujuk pada ekspresi linear disusun, digabungkan, dan bit yang hilang pada akhirnya dicari dengan menggunakan exhaustive search untuk tiap pasangan kandidat subkey hingga kunci yang benar ditemukan

1: **for** linear expressions $\mathcal{L}_1$ and $\mathcal{L}_2$ **do**
2: Sort the $C_{\hat{k}_i}$'s by decreasing $\left|\frac{N}{2} - C_{\hat{k}_i}\right|$ and rename them $C^*_j, 1 \leq j \leq 2^{12}$.
3: **for** $1 \leq j \leq 2^{12}$ **do**
4: /* ϵ is defined in Section 2 (expected bias of $\mathcal{L}$) */
5: **if** $\left(C^*_j - \frac{N}{2}\right)\epsilon > 0$ **then**
6: Guess $\mathsf{K}_{[k_1,\ldots,k_t]} = 0$
7: **else**
8: Guess $\mathsf{K}_{[k_1,\ldots,k_t]} = 1$
9: **end if**
10: **end for**
11: **end for**
12: Form 2^{24} $(C^*_i, C^*_j)_r$ pairs where $r := i \cdot j$.
13: Sort them by increasing r and rename them $D_k, 1 \leq k \leq 2^{24}$.
14: **for** $1 \leq k \leq 2^{24}$ **do**
15: Fix the key bits given by D_k and search exhaustively the remaining 30 bits of K until the right key is found.
16: **end for**

Kompleksitas C berhubungan dengan jumlah enkripsi DES yang diperlukan dalam proses exhaustive search dengan tingkat probabilitasnya adalah Pc, di mana kompleksitas C juga berhubungan juga dengan nilai kesuksesan saat menebak hasil dari kedua ekspresi linear.

Berdasarkan uji coba yang dilakukan oleh Matsui, maka didapatkan bahwa terdapat pola nilai probabilitas dari pola ekspresi dan juga fungsi F pada DES hingga mencapai 20 putaran



Serangan kriptanalisis linear terhadap DES, kecuali bagian exhaustive search, telah diimplementasikan di bagian sebelumnya. Setelah menentukan rank dari kandidat subkey pada list terakhir, tidaklah akan sulit untuk melakukan komputasi terhadap kompleksitas yang diharapkan dari bagian exhaustive search

$$\mathrm{E}[\hat{C}] = (r-1)\cdot 2^{30} + 2^{29}$$

Di mana r adalah rank dalam list D pada kandidat subkey. Kompleksitas perkiraan error memiliki nilai maksimum sebanyak 2^{99} evaluasi DES.

Tingkat kesuksesan probabilitas Pc dengan diketahui kompleksitas C juga bergantung pada probabilitas error saat melakukan penebakan bit pada K[k1…kt]. Probabilitas error didapat dengan

$$p_{wg} = \Pr\left[\text{“}\mathsf{K}_{[k_1,\ldots,k_t]} \text{ wrongly guessed”}\right] = \Phi_{(\mu_r,\sigma_r^2)}\left(\frac{N}{2}\right)$$

Tabel berikut memberikan perkiraan numeric untuk nilai N

N	2^{43}	$2^{42.5}$	2^{42}	2^{41}	2^{40}
p_{wg}	0.0004	0.0023	0.0086	0.0462	0.1170

3. Kesimpulan

Dari hasil studi dan analisis terhadap DES dan serangan kriptanalisis linear terhadap DES di atas, maka dapat ditarik beberapa kesimpulan:

1. Metode kriptanalisis linear merupakan metode serangan pertama terhadap DES yang menghasilkan plainteks keseluruhan 16 putaran DES
2. Ide dasar kriptanalisis linear adalah memperkirakan operasi dari sebagian dari chipertext yang memiliki ekspresi yang linear dengan operasi XOR.
3. Tujuan utama dari kriptanalisis linear yang diusulkan oleh Mitsuru Matsui adalah untuk menemukan pola yang berupa ekspresi linear pada algoritma cipher

Daftar Pustaka

Heys, Howard M. *A Tutorial on Linear and Differential Cryptanalysis*. Newfoundland: Memorial University of Newfoundland.

Junod, Pascal. 2001. *On The Complexity of Matsui's Attack.*

Mitsuru, Matsui. 1994. *Linear Cryptanalysis Method for DES Chiper*. Kanagawa: Mitsubishi Electric Corporation.

Munir, Rinaldi. 2006. *Kriptografi*. Bandung: Program Studi Teknik Informatika.

http://en.wikipedia.org/wiki/Data_Encryption_Standard. Tanggal akses: 31 Maret 2009 pukul 23.02

http://en.wikipedia.org/wiki/Linear_cryptanalysis Tanggal akses: 11 Maret 2009 pukul 13.32

http://searchsecurity.techtarget.com/sDefinition/0,,sid14_gci213893,00.html Tanggal akses: 12 Maret 2009 pukul 19.03

http://www.laynetworks.com/des.htm Tanggal akses: 11 Maret 2009 pukul 13.39

http://www.itl.nist.gov/fipspubs/fip46-2.htm Tanggal akses: 12 Maret 2009 pukul 19.11