Penerapan Mode Blok Cipher CFB pada Yahoo Messenger

Sesdika Sansani -- 13507047[1]
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
[1]*if17047@students.if.itb.ac.id*

***Abstraksi*— Instant Messenger (IM) adalah salah satu pelengkap dalam berkomunikasi atau bersosialisasi secara *online*. IM yang paling banyak digunakan di Indonesia adalah Yahoo Messenger (YM). Sebelum pesan yang dikirimkan pengirim diterima oleh penerima, pesan ini dikirimkan terlebih dahulu ke server dalam keadaan sebenarnya (plainteks) sehingga berpotensi disadap oleh orang lain dan ini dapat mengganggu privasi pengguna.**

Salah satu solusi yang dapat digunakan adalah dengan mengenkripsi pesan sebelum dikirimkan ke penerima. Salah satu metode enkripsi yang dapat digunakan adalah dengan mode blok cipher CFB 8-bit. Salah satu keunggulan dari CFB adalah hasil enkripsi pada suatu unit memberikan pengaruh pada hasil enkripsi unit selanjutnya sehingga jika terjadi suatu kesalahan di proses enkripsi atau dekripsi, dapat mempengaruhi hasil enkripsi atau dekripsi selanjutnya. Selain itu, hasil yang didapatkan menjadi acak sehingga menyulitkan kriptanalis untuk memecahkannya.

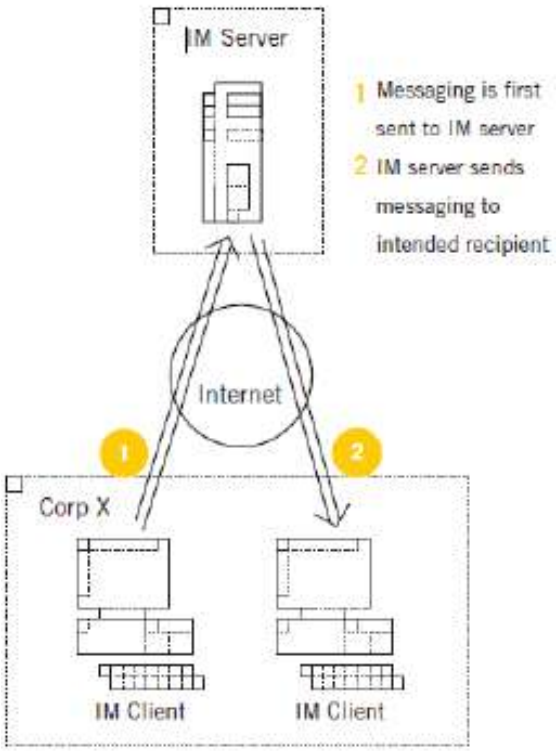
Index Terms*—Yahoo Messenger, *hacker*, plugin, *Cipher_Feedback

I. PENDAHULUAN

Instant Messenger (IM) adalah salah satu pelengkap dalam berkomunikasi atau bersosialisasi secara *online*. Selain *one to one*, *instant messenger* juga bisa digunakan ke suatu Group atau dikirimkan bersamaan ke beberapa orang dalam satu group, atau bahkan untuk *conference* atau *meeting online*. Banyak produk *instant messenger* yang dipergunakan di komputer ataupun perangkat smartphone, antara lain Yahoo Messenger, Google Talk, MSN Messenger atau Windows Live messenger. Yang paling banyak digunakan di Indonesia adalah Yahoo Messenger atau sering disingkat YM,

Gambar 1 Tampilan Aplikasi Yahoo Messenger

Pesan yang dikirimkan melalui YM adalah berupa plainteks. Pesan ini dikirimkan terlebih dahulu ke suatu server pusat sebelum diteruskan kepada pengirim yang dituju. Hal ini memberikan peluang untuk dapat menyadap pesan-pesan yang dikirimkan melalui YM. Tentu saja hal ini dapat mengganggu privasi penggunanya. Untuk lebih jelasnya, gambar berikut menunjukkan arsitektur YM secara lebih jelas.



Gambar 2 Arsitektur Yahoo Messenger!

Dari gambar di atas, terlihat bahwa pesan yang dikirimkan oleh suatu klien (sebagai pengirim) melewati server dahulu sebelum diterima oleh klien lainnya (sebagai penerima).

Karena itu, bagaimana cara mencegah orang-orang yang tidak bertanggung jawab seperti *hacker* untuk dapat membaca pesan yang dikirimkan melalui YM?

Salah satu solusi yang dapat digunakan adalah dengan melakukan enkripsi pada pesan yang dikirimkan oleh pengirim sebelum dikirimkan ke server. Salah satu metode

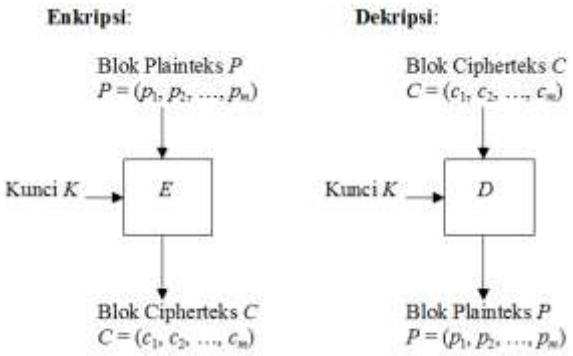
kriptografi yang dapat digunakan adalah dengan menggunakan blok cipher CFB.

II. PENERAPAN MODE BLOCK CIPHER CFB PADA YM

A. Block Cipher [1]

Dalam kriptografi, blok cipher adalah cipher simetrik yang beroperasi kunci pada kelompok bit yang panjangnya tetap yang disebut sebagai blok, dengan transformasi sebangun. Sebuah algoritma enkripsi cipher blok misalnya membutuhkan blok 128-bit plainteks sebagai input, dan output cipherteks blok 128-bit yang sesuai. Transformasi yang serupa dikendalikan menggunakan input kedua - kunci rahasia. Dekripsi adalah sama: dalam contoh ini algoritma dekripsi memerlukan sebuah blok 128-bit cipherteks bersama dengan kunci rahasia, dan menghasilkan blok 128-bit asli dari plainteks.

Gambar berikut ini merupakan skema enkripsi dan dekripsi pada blok cipher secara umum.



Gambar 3 Skema enkripsi dan dekripsi pada cipher blok

Sebuah cipher blok terdiri dari dua algoritma berpasangan, satu untuk enkripsi, E, dan yang lainnya untuk dekripsi, E^{-1}. Kedua algoritma ini menerima dua masukan: input blok ukuran n-bit dan kunci ukuran k-bit, menghasilkan sebuah blok output n-bit. Untuk setiap kunci yang sama, fungsi dekripsi adalah kebalikan dari enkripsi, sehingga

$$E_k(M) = C;$$

$$E_k^{-1}(C) = M$$

untuk setiap M blok dan kunci K. M disebut sebagai plainteks dan C sebagai cipherteks.

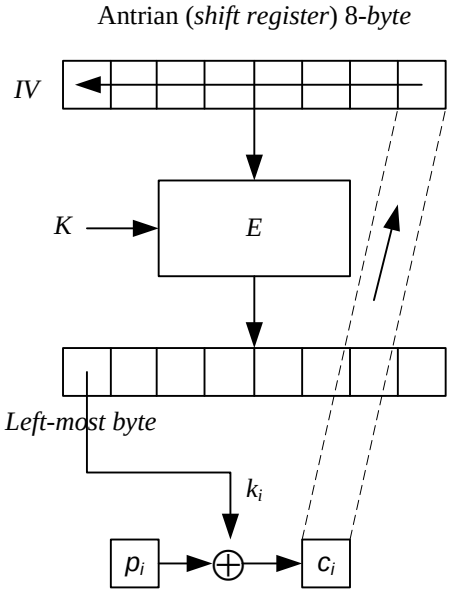
Untuk membuat algoritma blok cipher, ada beberapa mode yang dapat digunakan, salah satunya adalah metode CFB (*Cipher-Feedback*).

B. CFB (Cipher-Feedback)

Mode blok cipher ini memiliki rancangan bahwa jika terdapat suatu perubahan kecil pada hasil suatu operasi akan membawa dampak pada operasi berikutnya. Hal ini dapat menyulitkan kriptanalis untuk memecahkan cipherteks yang dia dapatkan dikarenakan hampir tidak ada pola yang ditimbulkan pada cipherteks yang dihasilkan. Selain itu, jika terjadi kesalahan pada proses enkripsi atau dekripsi, dapat mengakibatkan perubahan pada proses enkripsi atau dekripsi selanjutnya.

[2] Data yang dienkripsikan dengan mode CFB adalah dalam unit yang lebih kecil daripada ukuran blok. Unit yang dienkripsikan dapat berupa bit per bit (jadi seperti *cipher* aliran), 2 bit, 3-bit, dan seterusnya. Bila unit yang dienkripsikan satu karakter setiap kalinya, maka mode *CFB*-nya disebut *CFB* 8-bit.

CFB n-bit mengenkripsi plainteks sebanyak *n* bit setiap kalinya, $n \leq m$ (m = ukuran blok). Mode *CFB* membutuhkan sebuah antrian (*queue*) yang berukuran sama dengan ukuran blok masukan. Tinjau mode *CFB* 8-bit yang bekerja pada blok berukuran 64-bit (setara dengan 8 *byte*) pada gambar berikut.



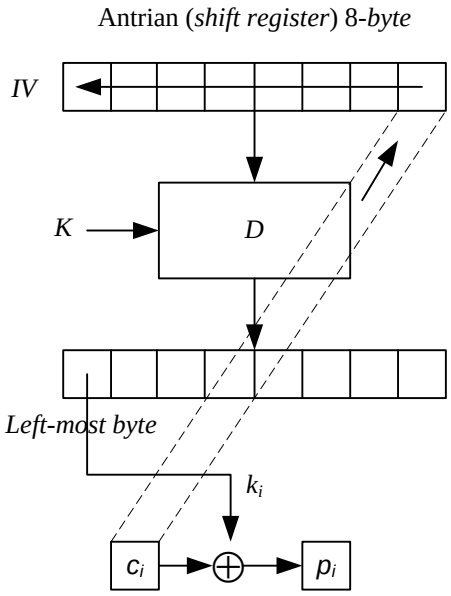
(a) Enkripsi

Gambar 4 Skema enkripsi pada mode CFB

Pada gambar 4, antrian register 8-byte pada awalnya merupakan *Inisialization Vector* (*IV*). *IV* dan *K* diproses dalam blok cipher *E* dan menghasilkan hasil antara. Unit dalam blok paling kiri akan di-*XOR* dengan unit plainteks. Unit cipherteks yang dihasilkan akan menjadi unit paling kanan dari antrian sebelumnya yang telah digeser ke kiri sebanyak satu unit.

Pada gambar 5, antrian register 8-byte pada awalnya juga merupakan *Inisialization Vector* (*IV*). *IV* dan *K* diproses dalam blok cipher *D* dan menghasilkan hasil antara. Unit dalam blok paling kiri akan di-*XOR* dengan unit cipherteks. Unit plainteks yang dihasilkan akan menjadi unit paling kanan. Unit yang menjadi unit paling kanan dari antrian sebelumnya yang telah digeser ke kiri

sebanyak satu unit adalah unit cipherteks.



(b) Dekripsi

Gambar 5 Skema dekripsi pada mode CFB

Dengan melihat skema di atas, pada dasarnya blok cipher E dan blok cipher D adalah sama sehingga tidak perlu membuat keduanya.

Secara formal, mode *CFB n*-bit dapat dinyatakan sebagai:

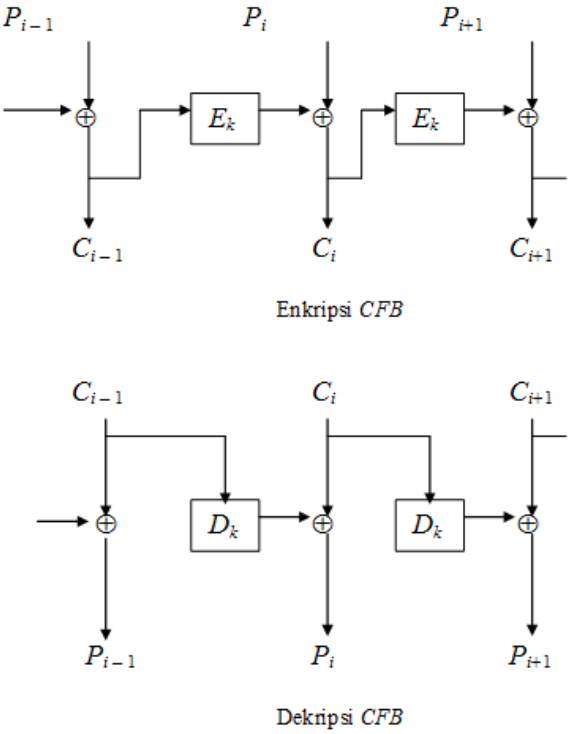
Proses Enkripsi: $C_i = P_i \oplus MSB_m(E_K(X_i))$
$X_{i+1} = LSB_{m-n}(X_i) \,||\, C_i$

Proses Dekripsi: $P_i = C_i \oplus MSB_m(D_K(X_i))$
$X_{i+1} = LSB_{m-n}(X_i) \,||\, C_i$

yang dalam hal ini,

X_i = isi antrian dengan X_1 adalah *IV*
E = fungsi enkripsi dengan algoritma *cipher* blok.
K = kunci
m = panjang blok enkripsi
n = panjang unit enkripsi
|| = operator penyambungan (*concatenation*)
MSB = Most Significant Byte
LSB = Least Significant Byte

Jika *m* = *n*, maka mode *CFB n*-bit adalah pada gambar 6 di bawah ini.

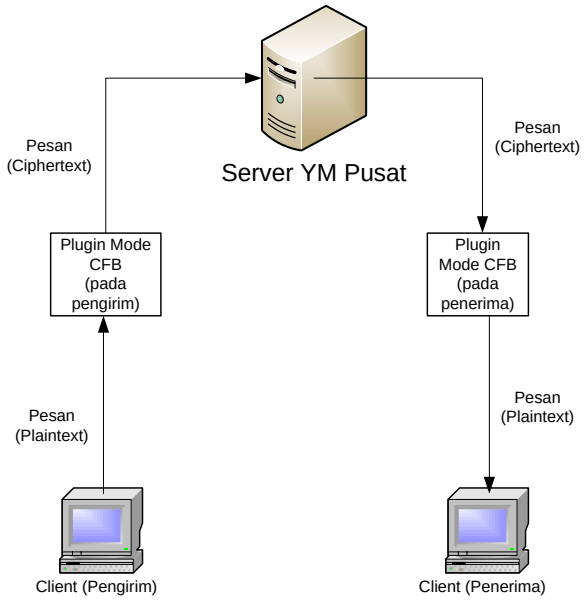


Gambar 6 Skema enkripsi dan dekripsi mode CFB m = n

III. PENERAPAN MODE CFB UNTUK MENGAMANKAN PESAN PADA YM

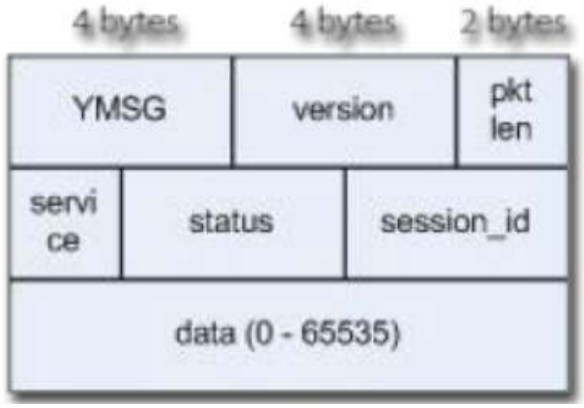
Salah satu solusi yang dapat digunakan untuk mengamankan pesan pada aplikasi Yahoo Messenger adalah dengan mengekripsi pesan yang dikirimkan oleh pengirim dan mendekripsi pesan yang diterima oleh penerima dengan menggunakan CFB 8-bit dengan panjang blok adalah 8-byte (selanjutnya akan disebut sebagai CFB saja sebagai CFB 8-bit). CFB akan diiimplementasikan sebagai plugin yang telah terintegrasi dengan YM. Untuk memudahkan pengguna, kunci yang digunakan adalah id ym salah satu komponen (bisa penerima atau pengguna, bergantung pada pengembang) sehingga pengguna tidak perlu memasukkan kunci lagi. IV yang digunakan dibangkitkan secara otomatis oleh plugin.

Skema dari enkripsi dengan mode CFB pada usulan untuk mengamankan pesan pada aplikasi YM ini adalah sebagai berikut:



Gambar 7 Skema Enkripsi-Dekripsi CFB yang diusulkan

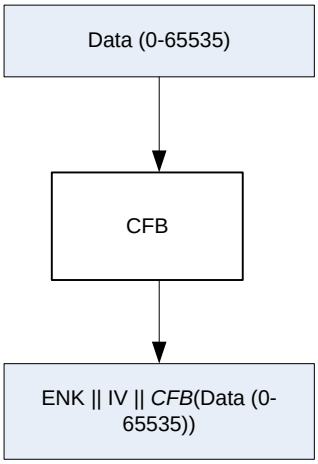
Enkripsi akan dilakukan pada plainteks YM, yaitu yang memiliki struktur paket berikut [3].



Gambar 8 Struktur paket pesan pada YM

Pada gambar di atas, dapat kita lihat bahwa paket pesan pada YM memiliki header sebagai tanda pengenal paket untuk aplikasi YM. Yaitu berupa id YMSG, versi aplikasi yang digunakan pengirim, panjang maksimal data pesan yang dapat dikirimkan, layanan yang digunakan, status, id dari session, dan data pesan itu sendiri.

Pada saat proses CFB dilakukan, plugin CFB yang diusulkan tidak akan mengubah *header* dari paket pesan dikarenakan *header* tersebut tetap diperlukan agar dapat dikirimkan ke penggunanya. Hasil dari CFB adalah karakter "ENK" akan digabungkan (menggunakan simbol ||) dengan IV yang dibangkitkan secara otomatis dan acak oleh plugin dan terakhir digabungkan dengan cipherteks yang dihasilkan. Hasil dari CFB adalah sebagai berikut:



Gambar 9 Enkripsi yang diusulkan

Kunci yang digunakan oleh blok cipher adalah dari salah satu id YM. Dimisalkan pengembang memilih id YM pengirim sebagai kunci. Dikarenakan panjang block adalah 8 byte, maka kunci yang dibutuhkan pun adalah sepanjang 8 byte (8 karakter). Jika id YM pengirim memiliki lebih dari 8 karakter, maka karakter yang diambil hanya 8 karakter pertama. Namun jika id YM pengirim kurang dari 8 karakter, akan ditambahkan karakter 'z' hingga panjang kuncinya menjadi 8 karakter.

Untuk proses dekripsi, digunakan lagi CFB yang sama, namun masukannya adalah cipherteks dan keluarannya adalah plainteks. Plugin akan mengecek terlebih dahulu apakah pesan yang diterima perlu didekripsi dengan membaca 3 karakter pertama dari pesan. Plugin akan melakukan dekripsi jika yang terbaca adalah karakter "ENK". Kemudian plugin akan membaca IV yang digunakan. Setelah itu, plugin akan melakukan proses dekripsi dan menampilkan hasilnya ke layar penerima.

III. Solusi yang Sudah Ada

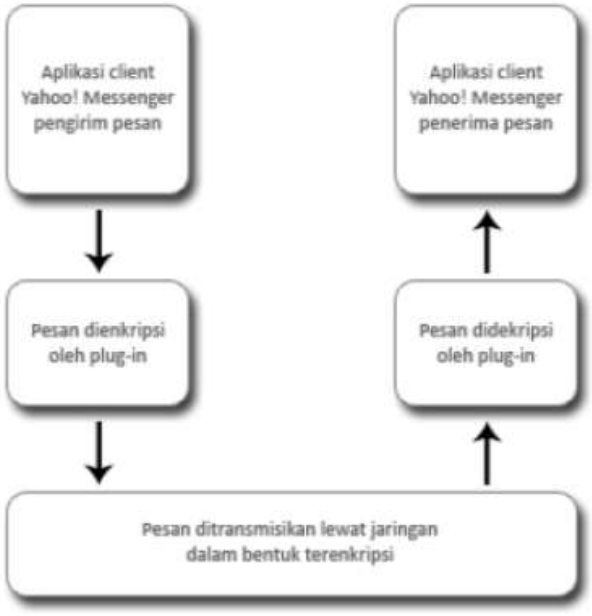
Ada beberapa solusi untuk permasalahan di atas, yaitu sebagai berikut:

A. Penggunaan Algoritma RSA pada Aplikasi Yahoo Messenger![3]

Salah satu solusi yang pernah muncul untuk permasalahan yang sama adalah dengan menggunakan algoritma RSA. Solusi ini dikemukakan oleh Mohamad Irvan Faradian dan David Susanto. Solusi yang dikemukakan adalah sebagai berikut:

- Enkripsi pesan yang ditransmisikan dari *client* (*sender*)
- Dekripsi pesan yang diterima client (*receiver*).
- Program enkripsi "ditanam" pada masing-masing aplikasi YM pengirim dan penerima.
- Ditanam sebagai program *add-in* atau *plug-in*.

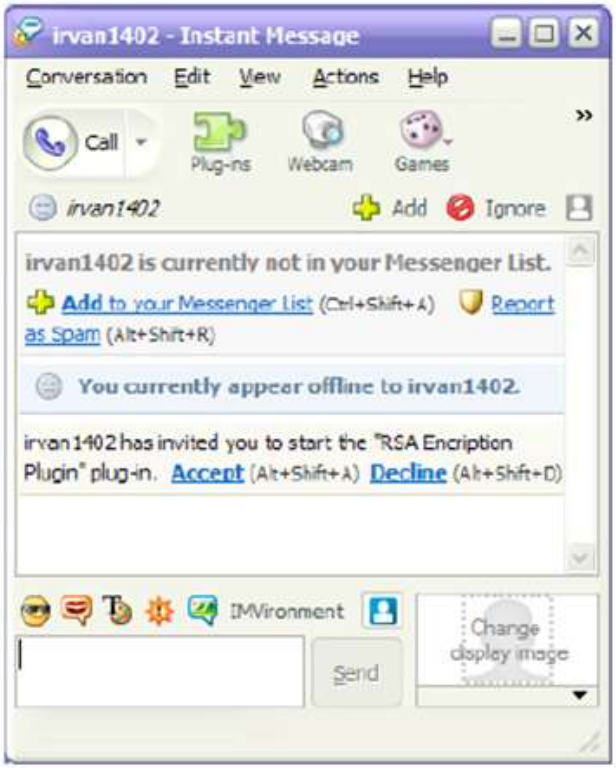
Skema dari cara kerja plugin yang dipaparkan adalah ada pada gambar berikut:



Gambar 10 Cara kerja plugin RSA

Sebagaimana yang telah dijelaskan sebelumnya bahwa pesan yang dikimkan oleh pengirim akan dienkripsi sebelum tiba di server, kemudian cipherteks tersebut tiba di server dan diteruskan ke penerimya. Sebelum sampai ke penerima, pesan akan didekripsi dahulu oleh plugin RSA agar dapat dibaca pengguna.

Untuk menggunakan plugin RSA ini, pengirim perlu mengundang penerima terlebih dahulu.



Gambar 11 Penawaran penggunaan plugin RSA

Penerima harus mengklik tombol “Accept” terlebih dahulu agar plugin dapat digunakan. Kemudian plugin akan menampilkan jendela untuk pengguna memasukkan Encryption exponent, module, pesan, dan setelah itu mengirimkannya. Gambar jendela plugin RSA pengirim adalah pada gambar 13.

Sedangkan pada penerima, akan ada jendela khusus baginya untuk memasukkan Decryption exponent dan modulnya agar dapat mendekripsi pesan yang diterima. Gambar jendela plugin RSA untuk penerima yaitu pada gambar 14.

Jika pesan hasil plugin RSA tersebut ditangkap dengan aplikasi *sniffer*, maka pesan yang terlihat adalah sebagai berikut:

```
14/01/2008 2:07:51      Yahoo:      irvan1402->
irvan140287 00dd9bbc29980780b8f3daae38519ff4
0120c00a1794229cc6c6092a448fdb9b
00072a80b4ffc2beb39cc97fd36c36b2
01a53e4e44c246f514ed950720f50a89
14/01/2008 2:07:53      Yahoo:
      216.155.193.161->irvan140287
      00dd9bbc29980780b8f3daae38519ff4
0120c00a1794229cc6c6092a448fdb9b
00072a80b4ffc2beb39cc97fd36c36b2
01a53e4e44c246f514ed950720f50a89
```

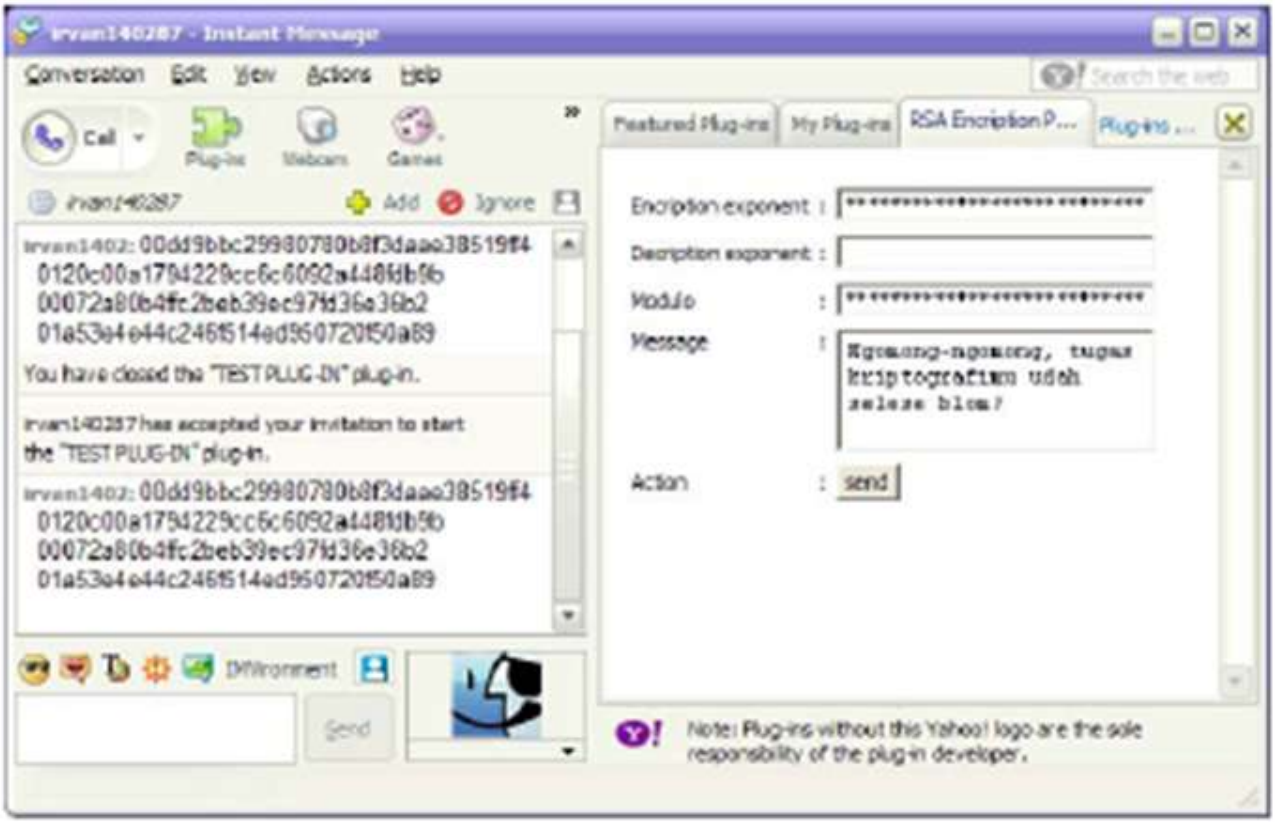
Gambar 12 Hasil capture isi cipherteks hasil plugin RSA dengan sniffer

Dapat dilihat pada gambar 12 bahwa hasil tangkapannya hanyalah kode-kode yang tidak memiliki arti yang dapat dipahami manusia.

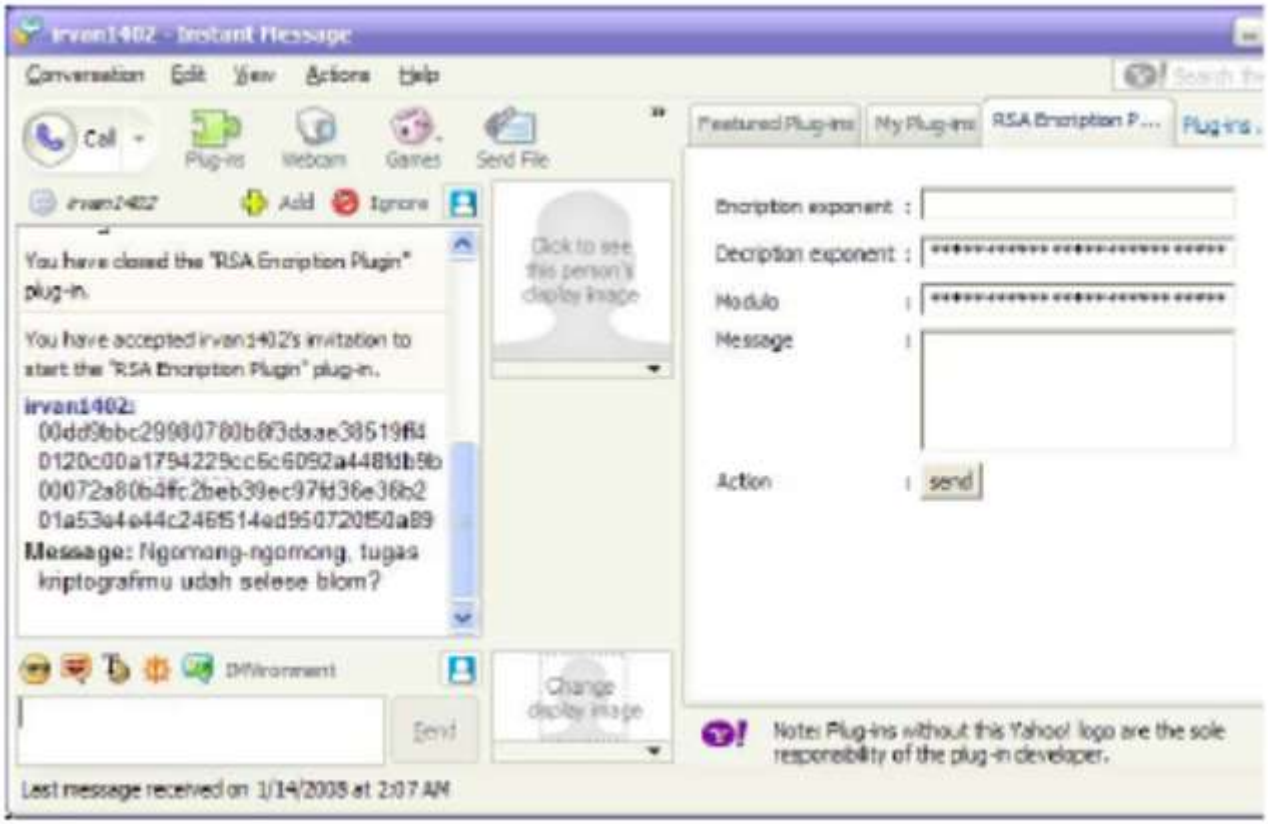
IV. KEUNGGULAN PLUGIN CFB YANG DIUSULKAN

Berikut adalah keunggulan dari plugin CFB yang diusulkan dibandingkan dengan plugin RSA yang pernah dipaparkan:

1. Hasil enkripsi unit-unit pada enkripsi blok mempengaruhi enkripsi unit-unit selanjutnya sehingga kriptanalis sulit untuk menemukan pemecahan dari cipherteks yang ditemukan.
2. CFB dirancang sedemikian rupa sehingga pengguna tidak perlu lagi memasukkan informasi lain berupa kunci dikarenakan plugin menggunakan salah satu id YM dan membangkitkan IV secara otomatis. Dengan begitu, pengguna lebih mudah dalam menggunakan plugin ini.
3. Tidak perlu ada kunci yang disimpan oleh pengirim maupun penerima pesan keamanan lebih terjamin. Memang kunci yang digunakan pada plugin CFB terbuka secara umum, namun hal ini dapat diakali dengan melakukan permutasi atau substitusi pada kunci sehingga kunci menjadi semakin acak.



Gambar 13 Penggunaan plugin RSA pada pengirim



Gambar 14 Penggunaan plugin RSA pada penerima

V. KESIMPULAN

Kesimpulan dari makalah ini adalah sebagai berikut:

1. Salah satu metode yang dapat digunakan untuk mengamankan pesan yang dikirimkan melalui YM adalah dengan mengenkripsi pesan tersebut.
2. Salah satu metode kriptografi yang dapat digunakan adalah dengan mode blok cipher CFB 8-bit. Kunci yang digunakan adalah id YM (bisa id pengirim atau penerima) dan nilai IV yang dibangkitkan oleh plugin secara otomatis sehingga pengguna tidak perlu lagi memasukkan kunci atau IV yang diinginkan.
3. Sudah ada solusi lain yang dipaparkan untuk permasalahan yang sama, yaitu algoritma RSA.
4. Keunggulan dari CFB yang diusulkan dibandingkan dengan RSA adalah hasil enkripsi unit dengan CFB dapat mempengaruhi hasil enkripsi berikutnya sehingga dapat menghilangkan pola dari hasil enkripsi; solusi dirancang agar pengguna tidak perlu repot untuk memasukkan kunci atau IV lagi; keamanan lebih terjamin dikarenakan tidak perlu ada tukar menukar kunci di antara pengirim dan penerima, adapun kunci dari id YM pengirim dan penerima dapat dikaburkan dengan melakukan substitusi dan permutasi padanya.

REFERENCES

[1] http://en.wikipedia.org/wiki/Block_cipher

[2] http://en.wikipedia.org/wiki/Block_cipher_modes_of_operation#Cipher_feedback_.28CFB.29

[3] http://webmail.informatika.org/~rinaldi/Kriptografi/2010-2011/Algoritma%20Kriptografi%20Modern_bag2%20(baru).ppt

[4] http://webmail.informatika.org/~rinaldi/Kriptografi/2010-2011/Aplikasi%20Enkripsi%20pada%20YM.ppt

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, 23 Maret 2011

Sesdika Sansani
13507047