Analisis dan Implementasi Pengamanan Pesan pada Yahoo! Messenger dengan Algoritma RSA

Mohamad Irvan Faradian
Program Studi Teknik Informatika, STEI, ITB, Bandung 40132, email: if14024@students.if.itb.ac.id

Abstrak – *Masalah keamanan pesan merupakan salah satu masalah yang terdapat dalam sebuah komunikasi yang berlangsung melalui internet. Saat ini, banyak media yang digunakan untuk melakukan komunikasi melalui internet seperti email, instant messaging, dan VOIP. Instant messaging merupakan salah satu layanan yang perkembangannya sangat pesat di kalangan pengguna internet. Hal ini dikarenakan layanan instant messaging memiliki beberapa kelebihan diantara layanan komunikasi yang lainnya. Namun, di balik beberapa kelebihan yang dimiliki, beberapa aplikasi instant messaging masih memiliki masalah keamanan dalam pesan yang ditransmisikan melalui internet. Salah aplikasi instant messaging yang memiliki masalah tersebut ialah Yahoo! Messenger. Makalah ini akan menguraikan analisis tingkat keamanan aplikasi Yahoo! Messenger terhadap serangan dalam bentuk penyadapan sebelum dan setelah menggunakan pengamanan enkripsi dengan algoritma RSA. Analisis akan dilakukan dengan cara melihat isi paket data yang ditransmisikan selama komunikasi antar pengguna berlangsung.*

Kata kunci*: komunikasi, instant messaging, Yahoo! Messenger, penyadapan, RSA*

1. PENDAHULUAN

Pada zaman sekarang ini, perkembangan internet begitu pesat. Internet selain menawarkan layanan untuk mencari informasi, juga menawarkan layanan untuk komunikasi. Layanan komunikasi yang ditawarkan melalui internet diantaranya ialah email, instant messaging, dan VOIP. Layanan yang paling banyak digunakan sekarang ini oleh pengguna internet ialah email dan instant messaging.

Bagi kebanyakan pengguna internet, email telah menggantikan peran surat untuk melakukan komunikasi melalui tulisan. Hal ini dikarenakan layanan email memiliki beberapa kelebihan seperti pengiriman yang cepat, mudah, murah, dan relatif aman. Di sisi lain, terdapat layanan komunikasi yang juga berbasis tulisan yang sama-sama memiliki kelebihan dibandingkan dengan berkomunikasi melalui surat, yaitu instant messaging.

Layanan instant messaging memiliki kelebihan dibandingkan dengan layanan email. Kelebihan tersebut ialah instant messaging mampu membuat penggunanya melakukan komunikasi secara real time. Jika seseorang mengirim pesan melalui email, maka pengirim pesan tersebut tidak mengetahui apakah penerima pesan sedang menggunakan internet atau tidak. Pengirim pesan tidak akan tahu kapan pesan yang dikirim akan dibalas oleh penerima pesan. Pengirim pesan harus memberi tahu penerima pesan terlebih dahulu jika pesan yang dikirim ingin dibalas secepatnya, misalnya melalui telepon. Namun, dengan layanan instant messaging, pengirim pesan tidak perlu memberi tahu penerima pesan terlebih dahulu agar penerima pesan membalas secepatnya. Hal ini dikarenakan dengan menggunakan layanan instant messaging, pengirim pesan dapat mengetahui apakah penerima pesan sedang menggunakan internet atau tidak.

Di balik kelebihan yang dimiliki layanan instant messaging, layanan ini masih memiliki kelemahan. Salah satu kelemahan tersebut ialah tingkat keamanan pesan yang ditransmisikan melalui internet. Yahoo! Messenger ialah salah satu aplikasi instant messenger yang tidak melakukan enkripsi terhadap pesan yang ditransmisikan melalui internet seperti yang dilakukan oleh beberapa instant messenger lainnya seperti GAIM dan Google Talk. Hal ini memungkinkan pihak lain dapat melakukan penyadapan terhadap pesan yang ditransmisikan melalui internet antar pengguna instant messenger.

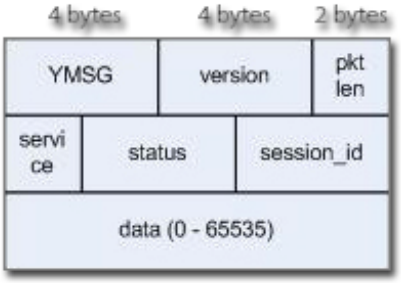
2. PENGENALAN SINGKAT YAHOO! MESSENGER

2.1. Arsitektur

Yahoo! Messenger termasuk aplikasi internet yang menggunakan arsitektur client-server. Ketika aplikasi client Yahoo! Messenger dijalankan, aplikasi ini akan mengirimkan pesan terlebih dahulu dengan server Yahoo! Messenger sebelum aplikasi client tersebut dapat bertukar pesan dengan aplikasi client yang lainnya. Hal ini mengharuskan pesan tersebut melalui jaringan internet yang menyebabkan aplikasi ini rawan terhadap serangan dalam bentuk penyadapan.

2.2. Protokol

Yahoo! Messenger menggunakan protokol yang dinamakan YMSG [1]. Struktur paket dan keterangan yang digunakan untuk melakukan komunikasi dapat dilihat pada gambar 1 [2].



Gambar 1. Struktur Paket Yahoo! Messenger

Berikut ini ialah penjelasan singkat struktur paket YMSG pada gambar 1:

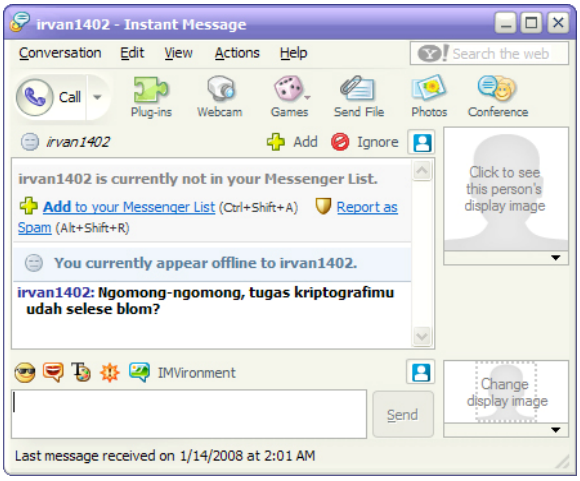
1) YMSG, 4 byte pertama dari seluruh paket selalu YMSG - nama protokol yang digunakan.
2) Version, 4 byte berikutnya ialah nomor versi protokol yang digunakan.
3) Pkt_len, 2 byte berikutnya menyatakan jumlah byte di dalam paket pada bagian data.
4) Service, menyatakan jenis service yang sedang diminta atau direspon.
5) Status, menyatakan status dari permintaan atau respon (sukses, gagal, atau yang lainnya).
6) Session_id, digunakan jika koneksi menggunakan proxy dan tidak digunakan jika menggunakan koneksi langsung.
7) Data, berisi pesan maupun file yang dikirimkan oleh pengguna.

3. ANALISIS KEAMANAN SEBELUM MENGGUNAKAN PENGAMANAN

Sebagai salah satu penyedia layanan instant messaging dalam skala besar, Yahoo! Messenger didesain untuk melayani jumlah pengguna yang sangat besar dari seluruh dunia [3]. Karena itu, aspek keamanan pesan yang ditransmisikan melalui internet kurang mendapat perhatian.

Berikut ini akan dilakukan pengujian tingkat keamanan Yahoo! Messenger versi 8 terhadap serangan dalam bentuk penyadapan sebelum menggunakan pengamanan. Pengujian akan dilakukan dengan cara membandingkan isi pesan asli yang dikirim masing-masing pengguna dengan isi paket data yang ditransmisikan melalui jaringan yang dilihat dengan menggunakan aplikasi packet sniffer.

Gambar 2 menunjukkan isi pesan asli yang dikirim masing-masing user. Gambar 3 menunjukkan isi paket yang ditransmisikan lewat jaringan yang ditangkap dengan menggunakan aplikasi packet sniffer (cari sumber). Isi paket tersebut merupakan isi paket yang dikirim oleh client ke server. Gambar 2 dan 3 menunjukkan bahwa pesan yang ditransmisikan lewat jaringan oleh aplikasi Yahoo! Messenger dari client ke server dan sebaliknya tidak aman. Hal tersebut dikarenakan isinya dapat disadap dan dimengerti makna pesannya.



Gambar 2. Isi Pesan

```
14/01/2008 2:01:10      Yahoo:      irvan1402->
irvan140287 □[1m<font face="Tahoma" size="8">Ngomong-
ngomong, tugas kriptografimu udah selese blom?
14/01/2008 2:01:11      Yahoo:      216.155.193.161->
irvan140287 □[1m<font face="Tahoma" size="8">Ngomong-
ngomong, tugas kriptografimu udah selese blom?
14/01/2008 2:01:20      WEB:  10.33.2.225->
http://f3.yahoofs.com/msgr/irvan1402/.friend_icon.png?
msTidkHBZqAL5Q7K
```

Gambar 3. Hasil Capture Isi Pesan dengan Sniffer

4. IMPLEMENTASI PENGAMANAN

4.1. Algoritma RSA

a. Langkah Pendahuluan

Sebelum memulai proses enkripsi dan dekripsi, ada beberapa langkah pendahuluan untuk membangkitkan beberapa nilai bilangan yang digunakan selama proses enkripsi dan dekripsi, yaitu:

1) Bangkitkan nilai dua buah bilangan prima yang besar, p dan q.
2) Hitung nilai n = p * q dan phi = (p - 1) * (q - 1).
3) Hitung nilai publik e yang nilainya $1 < e < phi$, sedemikian sehingga FPB(e, phi) = 1.
4) Hitung nilai privat d yang nilainya $1 < d < phi$, sedemikian sehingga e * d = 1 (mod phi).
5) Dengan demikian, kunci publik ialah pasangan (n, e) dan kunci privat ialah pasangan (n, d). Nilai p, q dan phi harus diharasiakan oleh pembangkit kunci.

b. Enkripsi

Enkripsi dilakukan oleh pengirim pesan dengan cara:

1) Ubah plainteks agar bisa direpresentasikan dalam bentuk bilangan m.
2) Hitung cipherteks $c = m^e \bmod n$ dengan nilai e dan n yang sudah didapatkan pada langkah pendahuluan.

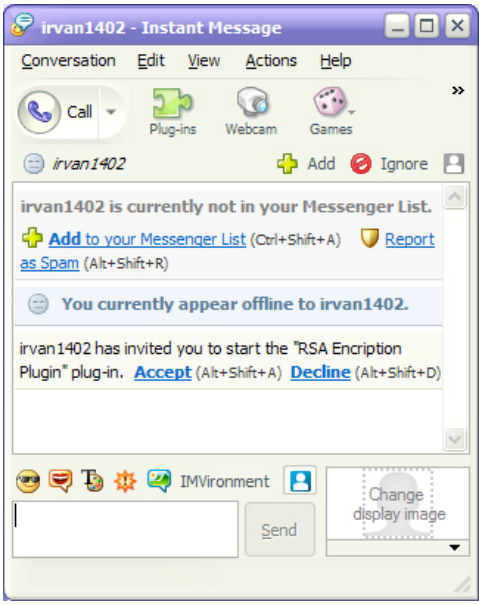
c. Dekripsi

Dekripsi dilakukan oleh penerima pesan dengan cara:

1) Hitung plainteks $m = c^d \bmod n$ dengan nilai d dan n yang sudah didapatkan pada langkah pendahuluan.
2) Ubah representasi bilangan m agar dapat menjadi plainteks semula.

4.2. Arsitektur dan Cara Kerja Plug-in

Plug-in yang digunakan untuk melakukan pengamanan transmisi pesan ini ditulis dalam bahasa javascript dipanggil dari file html. Plug-in ini dapat diluncurkan dari jendela percakapan sehingga kedua belah pihak yang menggunakan aplikasi client Yahoo! Messenger dapat melihat plug-in tersebut. Namun, sebelum dapat menggunakan plug-in secara bersama-sama, pihak yang diajak menggunakan plug-in ini akan dikirimi pesan oleh aplikasi server Yahoo! Messenger bahwa ada pihak lain yang mengajaknya menggunakan plug-in. Untuk lebih jelasnya, dapat dilihat pada gambar 4.

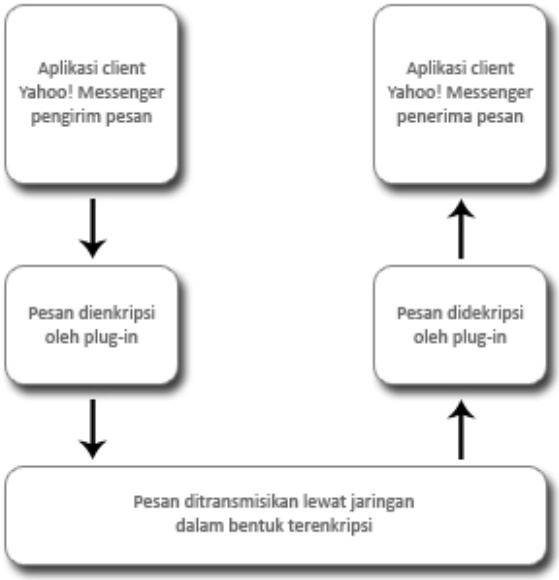


Gambar 4. Penawaran Penggunaan Plugin

Setelah pihak yang diajak menggunakan plug-in setuju untuk menggunakan plug-in, maka kedua belah pihak dapat memulai komunikasi dengan pesan yang terenkripsi dalam jaringan. Berikut ini langkah-langkah enkripsi dan dekripsi pada gambar 5 dalam proses komunikasi tersebut:

1) Pengirim pesan mengirim pesan.
2) Pesan tersebut dicegat oleh plug-in untuk dienkripsi dengan Algoritma RSA.
3) Pesan ditransmisikan oleh aplikasi client Yahoo! Messenger lewat jaringan dalam bentuk terenkripsi.
4) Pesan sampai di aplikasi client Yahoo! Messenger penerima pesan.
5) Pesan dicegat oleh plug-in untuk didekripsi dengan Algoritma RSA.
6) Penerima pesan menerima tampilan pesan yang sudah didekripsi.

Plug-in yang dibangun menerima masukan dari pengguna melalui sebuah form yang ditampilkan melalui tab plug-in pada jendela percakapan. Diasumsikan bahwa pengguna telah memiliki aplikasi pembangkit kunci, sehingga plug-in ini hanya menerima masukan nilai pasangan kunci publik (n, e) dan pasangan kunci privat (n, d).



Gambar 5. Cara Kerja Plugin

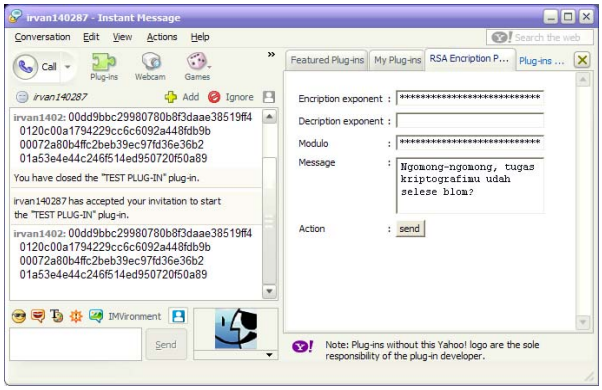
Perangkat lunak ini hanya terdiri dari tiga modul utama, yaitu:

1) File html script yang digunakan untuk membuat form yang akan menerima masukan nilai pasangan kunci publik (n, e) dan pasangan nilai kunci privat (d, e). Masukan yang ada dalam form ada empat, yaitu nilai n, nilai kunci publik, nilai kunci privat, dan pesan.
2) File javascript yang digunakan untuk memroses nilai pasangan kunci publik (n, e) dan pasangan kunci privat (d, e) yang dimasukkan pengguna untuk melakukan enkripsi dan dekripsi isi pesan. Javascript pada plug-in juga digunakan untuk memanggil API Yahoo! Messenger seperti untuk mencegat pesan sebelum ditransmisikan lewat jaringan dan mencegat pesan sebelum ditampilkan ke jendela percakapan. File javascript sendiri dipecah-pecah menjadi beberapa file, yaitu:
   a) Barrret.js, yang digunakan untuk mengoptimasi penghitungan dalam proses enkripsi dan dekripsi menjadi penghitungan modular agar proses enkripsi dan dekripsi menjadi lebih cepat.
   b) BigInt.js, yang digunakan untuk menampung nilai integer yang sangat besar.
   c) RSA.js, yang digunakan untuk melakukan enkripsi dan dekripsi dengan Algoritma RSA.
   d) Main.js, yang digunakan untuk menyatukan fungsi enkripsi, dekripsi, dan pemanggilan API Yahoo! Messenger.
3) File manifest yang berisi konfigurasi plug-in. File ini digunakan untuk mengetahui fungsi-fungsi yang terdapat dalam plug-in, lokasi direktori plug-in, dan jumlah pengguna yang diizinkan untuk menggunakan plug-in ini.

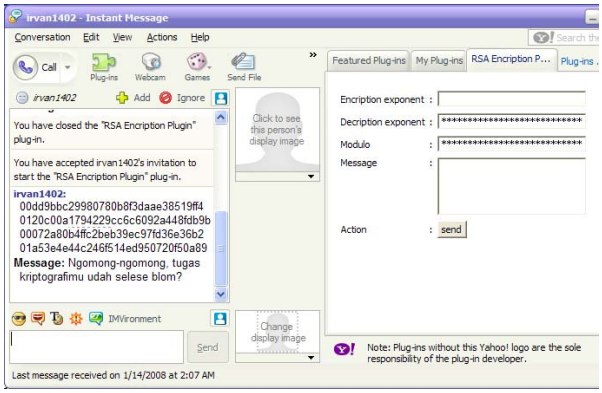
5. ANALISIS KEAMANAN DAN KECEPATAN TRANSMISI PESAN SETELAH MENGGUNAKAN PENGAMANAN

Pada dasarnya, pengamanan dalam proses transmisi pesan akan memperlambat proses transmisi pesan itu sendiri. Hal ini juga berlaku dalam proses komunikasi antar pengguna Yahoo! Messenger setelah menggunakan pengamanan dengan Algoritma RSA. Proses komunikasi akan mengalami delay karena dalam proses transmisi pesan tersebut, pesan akan dienkripsi terlebih dahulu ketika dikirim oleh pengirim pesan sebelum ditransmisikan melalui internet. Setelah pesan diterima, pesan tersebut harus didekripsi terlebih dahulu agar dapat dibaca oleh penerima pesan.

Berikut ini akan dilakukan pengujian kembali tingkat keamanan Yahoo! Messenger versi 8 terhadap serangan dalam bentuk penyadapan setelah menggunakan pengamanan dengan Algoritma RSA. Pengujian kembali dilakukan dengan cara membandingkan isi pesan asli yang dikirim masing-masing pengguna dengan isi paket data yang ditransmisikan melalui jaringan yang dilihat dengan menggunakan aplikasi packet sniffer.



Gambar 6. Penggunaan Plugin Pengirim



Gambar 7. Penggunaan Plugin Penerima

```
14/01/2008 2:07:51      Yahoo:      irvan1402->
irvan140287 00dd9bbc29980780b8f3daae38519ff4
0120c00a1794229cc6c6092a448fdb9b
00072a80b4ffc2beb39ec97fd36e36b2
01a53e4e44c246f514ed950720f50a89
14/01/2008 2:07:53      Yahoo:
      216.155.193.161->irvan140287
      00dd9bbc29980780b8f3daae38519ff4
0120c00a1794229cc6c6092a448fdb9b
00072a80b4ffc2beb39ec97fd36e36b2
01a53e4e44c246f514ed950720f50a89
```

Gambar 8. Hasil Capture Isi Pesan dengan Sniffer

Gambar 6 menunjukkan isi pesan asli yang dikirim masing-masing user. Gambar 8 menunjukkan isi paket yang ditransmisikan lewat jaringan yang ditangkap dengan menggunakan aplikasi packet sniffer (cari sumber). Isi paket tersebut merupakan isi paket yang dikirim oleh client ke server. Gambar 6 dan 8 menunjukkan bahwa pesan yang ditransmisikan lewat jaringan oleh aplikasi Yahoo! Messenger dari client ke server dan sebaliknya setelah menggunakan pengamanan menggunakan plug-in cukup aman. Hal tersebut dikarenakan walaupun dapat disadap, isi paket harus terlebih dahulu dideksipsi agar dapat dimengerti makna pesannya.

Selain melakukan pengujian terhadap tingkat keamanan setelah menggunakan pengamanan menggunakan plug-in, juga dilakukan pengujian pengaruh penggunaan pengamanan terhadap kecepatan transmisi pesan. Tabel 1 menunjukkan hasil pengujian kecepatan transmisi pesan menggunakan beberapa panjang kunci dengan isi pesan “Ngomong-ngomong, tugas kriptografimu udah selese blom?” yang panjangnya 54 karakter.

Tabel 1. Hasil Pengujian Panjang Kunci

Panjang kunci	Penambahan Kecepatan
32 bit	~ 0 detik
64 bit	~ 0 detik
128 bit	~ 1 detik
256 bit	~ 3 detik
512 bit	~ 10 detik
1024 bit	~ 77 detik

Pengujian pengaruh penggunaan pengamanan dengan plugin kali ini terhadap kecepatan transmisi pesan hanya dilakukan dengan cara mencoba beberapa panjang kunci. Sebenarnya, yang mempengaruhi kecepatan transmisi pesan dengan menggunakan plugin ini bukan hanya karena panjang kunci. Ada beberapa faktor lain, seperti panjang pesan, kecepatan internet, spesifikasi komputer, dan lain-lain. Namun, dalam pengujian diasumsikan bahwa percakapan dilakukan dalam keadaan normal, dengan panjang pesan sekitar 50 karakter.

6. KESIMPULAN

Yahoo! Messenger ialah salah satu instant messenger yang tidak mengenkripsi pesan yang ditransmisikan antar pengguna Yahoo! Messenger melalui internet. Hal ini mengakibatkan pesan yang ditransmisikan tersebut rawan terhadap serangan dalam bentuk penyadapan. Dari hasil analisis pengujian tingkat keamanan transmisi pesan yang telah dilakukan, isi pesan yang ditransmisikan dapat dengan mudah dibaca isinya. Isi pesan dapat dibaca dengan cara melihat isi paket data yang ditransmisikan melalui jaringan dengan menggunakan aplikasi packet sniffer.

Salah satu teknik yang dapat digunakan untuk melakukan pengamanan transmisi pesan pada Yahoo Messenger! ialah dengan menggunakan plug-in untuk mengenkripsi pesan. Plug-in yang digunakan ialah plug-in dengan tipe conversations, yaitu plug-in yang tersedia pada jendela percakapan. Dengan demikian, pengamanan transmisi pesan dapat dilakukan hanya kepada orang-orang tertentu, tidak kepada setiap orang yang diajak untuk melakukan percakapan. Dengan menggunakan plug-in ini, transmisi pesan lewat jaringan menjadi relatif lebih aman karena pesan yang dikirim oleh sebuah aplikasi client Yahoo! Messenger pengirim dienkripsi terlebih dahulu sebelum ditransmisikan lewat jaringan dan didekripsi setelah sampai pada aplikasi client Yahoo! Messenger penerima. Dari hasil analisis pengujian tingkat keamanan transmisi pesan yang telah dilakukan setelah menggunakan plug-in ini, isi pesan yang ditransmisikan tidak dapat dibaca dengan mudah. Pihak ketiga yang menyadap isi paket harus mendekripsi isi paket tersebut terlebih dahulu untuk bisa mengetahui makna pesan.

Namun, penggunaan plug-in ini memiliki kekurangan dalam hal kecepatan transmisi pesan itu sendiri. Karena algoritma yang digunakan untuk melakukan enkripsi dan dekripsi ialah Algoritma RSA, maka waktu yang digunakan untuk melakukan enkripsi dan dekripsi menjadi cukup lama jika panjang kunci yang digunakan lebih dari 512 bit. Hal ini mengakibatkan penerima pesan menunggu lebih lama untuk dapat melihat isi pesan jika dibandingkan dengan transmisi pesan biasa tanpa menggunakan plug-in.

7. DAFTAR REFERENSI

[1] http://www.astahost.com/yahoo-messenger-protocol-tutorial-part-8-signing-t11287.html

[2] http://libyahoo2.sourceforge.net/ymsg-9.txt

[3] http://www.symantec.com/avcenter/reference/secure.instant.messaging.pdf

[4] http://developer.yahoo.com/messenger/, Yahoo! Messenger Plug-in SDK User Guide and Reference

[5] B. Schneier, *Applied Cryptography: Protocols, Algorithms, and Source Code in C, 2nd Edition*, John Wiley & Sons, Inc, 1996.