Keamanan pada Layanan *Instant Messaging*: Studi Kasus Yahoo Messenger, Windows Live Messenger, dan Google Talk

Arie Karhendana
NIM 13503092

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

Abstrak. Saat ini, penggunaan *instant messaging* sebagai media komunikasi yang paling cepat pertumbuhannya sudah umum bagi pengguna internet. Layanan ini banyak dimanfaatkan untuk beragam keperluan, baik untuk bertukar pesan singkat dengan teman maupun untuk membicarakan masalah privat dengan rekan bisnis. Namun, belum banyak yang mengetahui sejauh mana tingkat keamanan layanan-layanan tersebut. Hal ini menjadi sangat penting, terutama jika informasi yang dipertukarkan adalah informasi yang sensitif dan rahasia.

Pada makalah ini akan dikaji sejauh mana tingkat keamanan beberapa layanan *instant messaging* publik terhadap serangan penyadapan (*eavesdropping*). Layanan *instant messaging* yang akan diuji sebagai studi kasus adalah Yahoo Messenger, Windows Live Messenger, dan Google Talk. Penilaian dilakukan dengan memeriksa isi paket data yang ditransmisikan selama proses komunikasi berlangsung.

Hasil pengujian menunjukkan bahwa layanan Yahoo Messenger dan Windows Live Messenger sama sekali tidak aman karena data yang ditransmisikan selama percakapan berupa plainteks dan dapat disadap dengan mudah. Sedangkan layanan Google Talk relatif lebih aman karena data yang ditransmisikan selama percakapan sudah dalam bentuk terenkripsi.

Kata kunci: *instant messaging*, penyadapan, *eavesdropping*, Yahoo Messenger, Windows Live Messenger, Google Talk

1 Pendahuluan

Tidak diragukan lagi pendapat yang menyatakan bahwa internet telah mengubah cara berkomunikasi. Bagi banyak orang, penggunaan email atau surat elektronik telah menggantikan surat konvensional dan bahkan sampai batas tertentu juga telah menggantikan telepon sebagai sarana berkorespondensi. Jutaan email dikirimkan setiap harinya dalam jaringan internet global. Email merupakan sarana komunikasi yang cepat dan mudah.

Namun di dalam dunia yang berputar sangat cepat seperti saat ini, terkadang bahkan penggunaan email masih dianggap kurang cepat dan kurang praktis. Sebagai contoh, pengirim email tidak dapat mengetahui apakah pada saat itu orang yang dikirimi email sedang *online* atau tidak. Sehingga, pengirim tidak bisa mengharapkan pesan emailnya dapat ditanggapi sesegera mungkin. Selain itu, jika seseorang berkorespondensi dengan orang lain secara terus menerus, orang tersebut biasanya harus melewati beberapa langkah untuk membaca, membalas, dan mengirim email. Oleh karena itu, *instant messaging* pun menjadi populer karena dianggap memiliki kelebihan-kelebihan dibanding email.

Instant messaging (lazim disingkat IM) adalah salah satu jenis layanan komunikasi yang memungkinkan seseorang untuk melakukan percakapan (*chat*) privat dengan orang lain secara *real time* melalui internet. Pada umumnya, percakapan melalui *instant*

messaging ini berupa pesan teks. Namun, bisa saja berupa pesan suara atau video.

Sistem *instant messaging* juga memungkinkan pengguna untuk menyimpan daftar orang yang dapat diajak berkomunikasi dalam sebuah daftar kontak (*contact list*) atau daftar teman (*buddy list*). Sistem akan memberikan informasi jika ada kontak yang sedang *online*. Sehingga pengguna dapat memulai percakapan dengan kontak tersebut dan kemudian saling bertukar pesan. Komunikasi pun terjadi secara instan dan *real time*. Fasilitas *instant messaging* ini disebut *presence information*.

1.1 Perkembangan *Instant Messaging*

Walaupun *instant messaging* tampak seakan-akan sebuah teknologi baru, sebenarnya *instant messaging* telah berusia lebih dari sepuluh tahun [1]. Beberapa literatur menyatakan bahwa layanan *instant messaging* yang pertama adalah layanan *instant messaging* yang diperkenalkan oleh AOL (American Online) pada awal tahun 1990-an [2]. Layanan ini merupakan fasilitas yang disediakan bagi para pengguna penyedia jasa internet terbesar di Amerika Serikat ini. Layanan *instant messaging* AOL memungkinkan para penggunanya untuk saling berkomunikasi dan bertukar pesan secara *online*. Namun, penggunanya baru terbatas hanya pelanggan AOL saja. Di kemudian hari, AOL memperkenalkan AOL Instant Messenger (lazim disingkat AIM) yang tidak dibatasi hanya untuk pelanggan AOL saja, sehingga memungkinkan setiap orang untuk menggunakan AIM.

Beberapa literatur lain menyatakan bahwa *instant messaging* telah ada dalam bentuk yang beragam jauh sebelum AIM dikembangkan [3]. Sistem *instant messaging* awal telah diimplementasikan pada jaringan komputer privat seperti misalnya Term-Talk pada sistem PLATO pada awal 1970-an. Sistem yang serupa juga telah diimplementasikan pada komputer DEC PDP-11 sebagai program talk. Sampai sekitar tahun 1990-an, program talk masih lazim digunakan di kalangan akademik untuk saling berkomunikasi dengan menggunakan sistem operasi UNIX/Linux. Kemudian, pada tahun 1987, MIT mengembangkan *instant messaging client* pertama yang berbasis grafis pada sistem Zephyr.

Pada tahun 1988, Jarkki Oikarinen mengembangkan sistem Internet Relay *Chat* (IRC). IRC memungkinkan pengguna untuk saling berkomunikasi dalam "ruang percakapan" (*chat room*) dan saling berkirim file. IRC menjadi sangat terkenal pada tahun 1990-an. Namun, tidak seperti sistem *instant messaging* modern, IRC tidak memiliki fasilitas untuk mengetahui apakah seorang kontak atau rekan sedang *online* atau tidak. Selain itu, pada IRC tidak ada keharusan untuk melakukan registrasi pengguna maupun batasan untuk menggunakan nama tertentu.

Perkembangan *instant messaging* yang sangat pesat dimulai sejak tahun 1996 ketika Mirabilis, sebuah perusahaan Israel, memperkenalkan ICQ (biasa dibaca I Seek You dalam bahasa Inggris) ke internet [4]. Sejak saat itu, penggunaan istilah "*instant messaging*" pun menjadi populer.

Hampir bersamaan dengan ICQ, AOL pun memperkenalkan AIM yang dapat digunakan tidak hanya oleh pelanggan AOL saja. Kedua layanan ini pun kemudian bersaing ketat. Namun, pada akhirnya AOL mengakuisisi Mirabilis dan layanan ICQ pada tahun 1998. Sehingga AIM & ICQ pun menjadi layanan *instant messaging* yang paling dominan saat itu.

Setelah itu, banyak layanan *instant messaging* bermunculan dengan fasilitas khasnya masing-masing. Yahoo sebagai *port*al internet terbesar saat itu memperkenalkan layanan terbarunya, yaitu Yahoo Messenger. Layanan ini terintegrasi dengan layanan Yahoo yang lain, seperti email dan kalender. Microsoft pun merilis layanan *instant messaging* yang terintegrasi dengan *port*al MSN dan layanan Hotmail, yaitu MSN Messenger. Layanan *instant messaging* yang paling terakhir dirilis adalah Google Talk, yang diperkenalkan oleh Google sebagai search engine terbesar saat ini. Google Talk terintegrasi dengan layanan Google lainnya, terutama layanan Gmail.

Selain layanan tersebut, terdapat beberapa layanan *instant messaging* lain yang ada. Misalnya QQ, Gadu-Gadu, dan Sametime. Bahkan penyedia jasa VoIP (*Voice over Internet Protocol*) terkenal, Skype, juga memiliki fasilitas *instant messaging* dalam sistemnya.

Masing-masing layanan tersebut menggunakan protokolnya sendiri. Kebanyakan protokol tersebut bersifat tertutup (*proprietary*). Selain itu, pengguna layanan yang satu tidak dapat saling berkomunikasi dengan pengguna

layanan yang lain. Oleh karena itu, ada usaha-usaha untuk mengembangkan protokol *instant messaging* yang terbuka dan dapat saling berkomunikasi. Saat ini, telah dikembangkan beberapa protokol *instant messaging* yang bersifat terbuka, seperti XMPP/Jabber dan SIP/SIMPLE.

Selain itu, para penyedia layanan *instant messaging* besar juga mulai mengembangkan cara agar para pengguna layanan mereka dapat berkomunikasi dengan pengguna layanan lain. Misalnya Yahoo dan Microsoft yang telah sepakat untuk menyediakan fasilitas interkoneksi, sehingga pengguna kedua layanan dapat saling berkomunikasi. Selain itu, Google dan AOL juga mengembangkan fasilitas serupa.

Di kalangan *enterprise* dan perusahaan, berkembang pula sistem *instant messaging* khusus untuk internal perusahaan. Perusahaan dapat memasang perangkat *instant messaging* privat yang tidak terhubung dengan layanan *instant messaging* publik lain untuk digunakan oleh karyawannya. Dengan adanya sistem ini, diharapkan perusahaan dapat meningkatkan produktivitas dan efisiensi dalam berkomunikasi.

Saat ini, *instant messaging* merupakan media komunikasi yang paling cepat pertumbuhannya. Pada akhir tahun 2006, diperkirakan terdapat 390 juta pengguna, baik individual maupun *enterprise* [2]. Dilaporkan terdapat 1 milyar pesan yang dikirimkan setiap harinya melalui layanan *instant messaging* besar seperti AIM, MSN Messenger, dan Yahoo Messenger. Diperkirakan pula, trafik *instant messaging* pada jaringan internet akan melebihi trafik email pada akhir tahun 2006.

1.2 Keuntungan *Instant Messaging*

Instant messaging memungkinkan efisiensi komunikasi dan memudahkan dalam berkolaborasi. Dibandingkan dengan email dan telepon, pengguna *instant messaging* dapat mengetahui jika seorang kontak atau rekannya dapat dihubungi (*available*). Kebanyakan sistem *instant messaging* juga memungkinkan pengguna untuk mengeset status *online* mereka sehingga teman-temannya dapat mengetahui apakah pengguna tersebut sedang *available*, sibuk (*busy*), atau pergi dari depan komputer (*away*). Di sisi lain, pengguna tidak harus menjawab setiap pesan yang masuk sesegera mungkin. Oleh karena itu, penggunaan *instant messaging* dianggap lebih sopan dan tidak ‘memaksa’ seperti layaknya telepon.

Selain itu, penggunaan *instant messaging* juga cocok untuk saling bertukar informasi singkat seperti alamat URL atau potongan teks dokumen. Hal-hal ini sulit dilakukan jika menggunakan media telepon.

Karena sifatnya yang langsung dan dua arah, banyak pengguna yang menganggap bahwa penggunaan *instant messaging* untuk pekerjaan memungkinkan untuk meningkatkan produktivitas. Oleh karena itu, *instant messaging* banyak digunakan pada lingkungan profesional pada lingkungan kantor.

Para penyedia layanan *instant messaging* pun berlomba-lomba untuk menambah fasilitas pada layanan mereka. Beberapa fasilitas yang lazim ditemukan pada *instant messaging* meliputi:

a. *Chat*
   Melakukan percakapan dengan kontak.
b. *Buddy list & presence information*
   Menyimpan daftar rekan yang biasa dihubungi serta mengetahui rekan mana saja yang sedang *online*.
c. *Conference*
   Melakukan percakapan dengan banyak orang sekaligus dalam satu sesi.
d. *File transfer*
   Memungkinkan untuk berkirim file dengan pengguna lain.
e. *History*
   Menyimpan daftar percakapan yang telah dilakukan.

Selain itu, beberapa layanan *instant messaging* juga memberikan nilai tambah berupa integrasi dengan layanan mereka yang lain, seperti:

a. Notifikasi email
   Memberi informasi jika ada email baru yang masuk ke mailbox pengguna.
b. *Content*
   Menampilkan informasi seperti berita terbaru, cuaca, harga saham, maupun jadwal dalam kalender.
c. *Voice* & video *chat*
   Bercakap-cakap dengan menggunakan media suara dan video, sehingga pengguna dapat melihat dan mendengar rekan yang sedang diajak berkomunikasi.

1.3 Aspek Keamanan *Instant Messaging*

Seiring dengan semakin banyak digunakannya layanan *instant messaging*, maka aspek *security* (keamanan) menjadi sangat penting

untuk dipertimbangkan. Apalagi jika data yang dikirimkan via layanan tersebut merupakan data yang sensitif dan rahasia. Tanpa mengesampingkan pengguna terbesar *instant messaging*, yaitu kalangan pribadi atau individual, keamanan merupakan masalah terbesar bagi pengguna *instant messaging* pada perusahaan atau *enterprise*.

Sayangnya, kebanyakan sistem *instant messaging* saat ini didesain bukan berdasarkan aspek keamanan sebagai pertimbangan utama, melainkan aspek skalabilitas untuk menunjang jumlah pengguna yang begitu besar [1]. Hampir semua layanan *instant messaging* publik tidak memiliki fasilitas enkripsi dan kebanyakan memiliki fasilitas untuk melewati (bypass) *firewall* perusahaan. Selain itu, *instant messaging* juga rentan terhadap penyebaran virus dan *worm* [5].

Symantec Enterprise Security menyatakan beberapa serangan yang rentan pada layanan *instant messaging* [1], yaitu:

a. Penyadapan (*eavesdropping*)
   Karena kebanyakan sistem *instant messaging* tidak mengenkripsi data yang dikirim, maka pihak ketiga bisa menyadap komunikasi yang berlangsung antara pengguna *instant messaging* dengan menggunakan packet *sniffer* atau teknologi sejenis.
b. Pembajakan account (account hijacking)
   Kebanyakan sistem *instant messaging* rentan terhadap pembajakan account atau penyamaran (*spoofing*). Seseorang dapat membajak account orang lain atau menyamar menjadi orang tersebut.
   Proteksi terhadap password pada sistem *instant messaging* juga sangat lemah. Terkadang password pengguna disimpan dalam komputer klien, baik dalam bentuk terenkripsi atau tidak. Walaupun tersimpan dalam bentuk terenkripsi, password pengguna masih dapat di-*crack* dengan bantuan kakas yang sesuai.
c. Pengaksesan & pengubahan data pengguna
   Sebagaimana perangkat lunak yang terhubung ke internet lainnya, program *instant messaging* dapat memiliki *bug* yang memungkinkan penyerangan melalui jaringan, seperti memanfaatkan *buffer overflow* atau paket data yang sudah dimodifikasi sedemikian rupa. Sehingga penyerang dapat mengambil alih komputer pengguna.
d. *Worm* dan serangan kombinasi
   Seperti email, *instant messaging* memungkinkan penyebaran *worm* dan virus. Contoh kasus nyata adalah *worm* pada jaringan IRC.

Pada makalah ini, akan dipaparkan sejauh mana tingkat keamanan layanan *instant messaging* dari serangan penyadapan (*eavesdropping*). Pengujian dilakukan berdasarkan pemeriksaan isi paket data yang dikirimkan melalui jaringan publik. Layanan *instant messaging* yang akan diuji dibatasi pada Yahoo Messenger, Windows Live Messenger, dan Google Talk.

2 Layanan *Instant Messaging* Publik

Kebanyakan layanan *instant messaging* bersifat publik. Setiap orang dapat dengan bebas mendaftar ke suatu layanan *instant messaging* dan kemudian men-*download* program *client* layanan tersebut dengan gratis.

Masing-masing penyedia layanan memberikan fasilitas yang beragam untuk memenuhi kebutuhan penggunanya. Daftar penyedia layanan *instant messaging* yang umum digunakan dapat dilihat pada Tabel 1.

Tabel 1: Beberapa layanan instant messaging publik

Layanan	Penyedia
AIM	AOL
ICQ	AOL
Yahoo Messenger	Yahoo
Windows Live Messenger[1]	Microsoft
Google Talk	Google

Kebanyakan pengguna suatu layanan tidak dapat saling berkomunikasi dengan pengguna layanan lain. Sehingga untuk berkomunikasi dengan pengguna layanan lain, seseorang harus memiliki *account* pada masing-masing layanan.

Namun beberapa penyedia layanan *instant messaging* sudah menjajaki untuk membuat interkoneksi dengan penyedia layanan lain, sehingga pengguna kedua layanan dapat saling

[1] Sebelumnya dikenal dengan nama MSN Messenger.

berkomunikasi. Contohnya adalah Windows Live Messenger dan Yahoo Messenger.

2.1 AOL Instant Messenger

AOL Instant Messenger (AIM) adalah layanan *instant messaging* yang disediakan oleh AOL. AIM pertama kali dirilis oleh AOL pada bulan Mei 1997. Saat ini AIM merupakan layanan *instant messaging* yang paling banyak digunakan, dengan sekitar 53 juta pengguna aktif [6].

2.2 ICQ

ICQ (sering dibaca I Seek You dalam bahasa Inggris) adalah layanan *instant messaging* yang disediakan oleh AOL. Sebelum dimiliki oleh AOL, ICQ dimiliki oleh Mirabilis yang merupakan pembuat ICQ. ICQ dirilis pertama kali pada November 1996.

Tidak seperti sistem *instant messaging* lain yang mengidentifikasi pengguna berdasarkan username atau email, pada ICQ, pengguna diidentifikasi berdasarkan serangkaian angka yang disebut UIN (Unified Identification Number). Saat ini, terdapat sekitar 20 juta pengguna aktif ICQ.

ICQ adalah layanan *instant messaging* pertama kali yang telah menginspirasi layanan lain. Walaupun layanan ini telah dimiliki oleh AOL, namun fasilitas untuk saling berkomunikasi dengan para pengguna AIM masih dalam tahap beta.

2.3 Yahoo Messenger

Yahoo Messenger adalah layanan *instant messaging* yang disediakan oleh Yahoo, salah satu *port*al terbesar di internet. Yahoo Messenger memiliki berbagai fasilitas yang terintegrasi dengan layanan Yahoo yang lain, seperti email, kalender, berita, cuaca, dan harga saham.

Yahoo Messenger dilengkapi dengan fasilitas *Voice Chat* dan Webcam, sehingga pengguna dapat berkomunikasi dengan suara dan video. Selain itu, pengguna Yahoo dapat mengalihkan pesan yang diterimanya ke ponsel melalui SMS (Short Message Service). Salah satu fitur yang khas pada Yahoo Messenger adalah IMvironments yang memungkinkan pengguna untuk mengubah tampilan jendela percakapan. Yahoo Messenger pun memiliki fitur avatar, sehingga pengguna dapat memilih gambar atau foto yang mewakili pengguna di dunia maya.

Saat ini, pengguna Yahoo Messenger dapat saling berkomunikasi dengan para pengguna Windows Live Messenger setelah ada kesepakatan antara pihak Yahoo dan Microsoft sebagai pengelola layanan.

Pengguna Yahoo Messenger diidentifikasi berdasarkan Yahoo ID, yaitu nama generik yang dapat digunakan pula untuk mengakses layanan Yahoo yang lain. Saat ini, pengguna aktif Yahoo Messenger berjumlah sekitar 22 juta orang [6].

2.4 Windows Live Messenger

Windows Live Messenger (sebelumnya dikenal dengan nama MSN Messenger) adalah layanan *instant messaging* yang dikelola oleh Microsoft. Layanan ini merupakan bagian dari layanan *online* Windows Live milik Microsoft.

Layanan ini pertama kali dirilis pada bulan Juli 1999. Saat ini jumlah pengguna aktifnya adalah 27 juta, sehingga menempatkan Windows Live Messenger pada urutan kedua berdasarkan jumlah pengguna aktif [6].

Program *client* Windows Live Messenger hanya tersedia untuk sistem operasi Windows. Dukungan untuk sistem operasi Macintosh sangat minim, bahkan tidak ada *client* layanan ini pada sistem operasi Linux. Walaupun demikian, terdapat beberapa program *client* pihak ketiga yang dapat digunakan untuk layanan ini, seperti GAIM dan aMSN.

Pengguna Windows Live Messenger dapat berkomunikasi dengan pengguna Yahoo Messenger, setelah ada kesepakatan bersama antara Microsoft dan Yahoo. Dengan demikian, pengguna Windows Live Messenger dapat melakukan percakapan atau menambahkan alamat pengguna Yahoo ke dalam daftar kontak mereka.

Windows Live Messenger mendukung komunikasi via suara dan video. Selain itu, layanan ini juga mendukung integrasi dengan layanan lain yang tergabung dalam naungan Windows Live. Pengguna Windows Live Messenger diidentifikasi berdasarkan Windows Live ID yang bisa didapatkan setelah melakukan registrasi melalui situs Microsoft Pass*port* Network.

2.5 Google Talk

Google Talk adalah layanan *instant messaging* yang disediakan oleh Google sebagai search engine terbesar di internet. Sampai saat ini, layanan ini masih berstatus beta.

Tidak seperti layanan *instant messaging* lain yang menggunakan protokol yang bersifat tertutup (*proprietary*), Google Talk menggunakan protokol XMPP/Jabber yang bersifat terbuka dan telah distandardisasi. Dengan demikian, pengguna layanan ini dapat berkomunikasi dengan pengguna layanan berbasis Jabber pada *server* lain. Lebih jauh lagi, pengguna tidak harus terpaku untuk menggunakan sebuah program *client* saja, tetapi dapat menggunakan program *client* lain yang berbasis Jabber. Hal ini berdasarkan pernyataan dari Google yang menyatakan bahwa tujuan Google Talk adalah interoperabilitas dengan layanan berbasis Jabber lainnya.

Google Talk terintegrasi erat dengan Gmail, layanan email dari Google. Sehingga pengguna dapat melakukan komunikasi Google Talk melalui antarmuka yang sama dengan Gmail. Selain itu, Google Talk juga memiliki fasilitas percakapan melalui suara. Namun, percakapan melalui suara hanya didukung melalui program *client* resmi yang dirilis oleh Google.

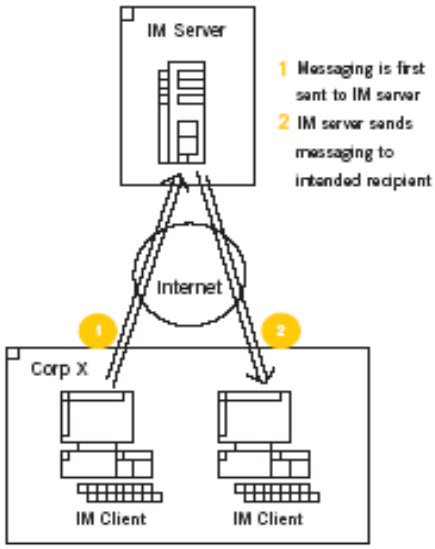
Pengguna Google Talk diidentifikasi berdasarkan account Gmail maupun account lain yang telah didaftarkan sebagai Google Account.

Google telah membuat kesepakatan dengan AOL untuk membuat interkoneksi antara layanan Google Talk dengan AIM. Sehingga saat ini, pengguna Google Talk dapat berkomunikasi langsung dengan pengguna AIM.

3 Cara Kerja *Instant Messaging*

3.1 Arsitektur

Pada umumnya, sistem *instant messaging* bekerja berdasarkan arsitektur *client-server*. Pengguna menjalankan *instant messaging client* pada komputernya dan kemudian program *client* tersebut akan berkomunikasi dengan *server instant messaging* yang telah disediakan oleh pengelola layanan untuk bertukar pesan atau informasi tentang pengguna dengan pengguna lain.



Gambar 1: Instant messaging dengan arsitektur client-server [1]

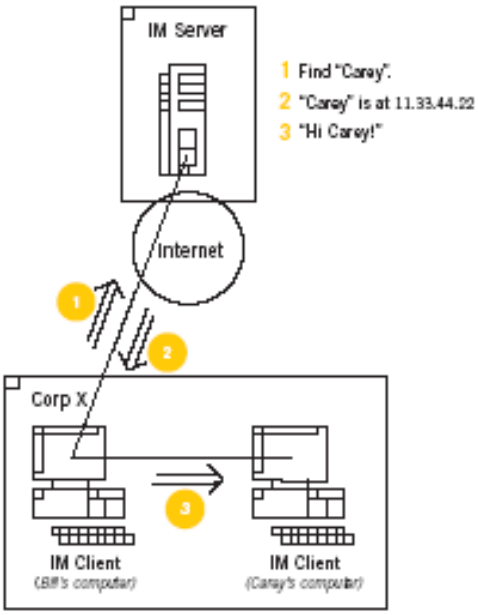
Pada kebanyakan sistem *instant messaging* berarsitektur *client-server*, pesan yang dipertukarkan di antara pengguna berupa plainteks. Pesan ini sangat rentan terhadap penyadapan di sepanjang jalur komunikasi dari *client* ke *server*. Hanya beberapa sistem *instant messaging* saja yang mengenkripsi pesan yang dikirimkan dari *client* ke *server* dan sebaliknya.

Dapat dilihat pada Gambar 1 bahwa walaupun kedua komputer berdekatan atau berada pada jaringan yang sama, namun trafik yang dikirimkan di antara kedua komputer tersebut harus melalui jaringan internet global. Dengan penggunaan kakas yang tepat, seseorang dapat menyadap komunikasi yang terjadi antara kedua komputer tersebut walaupun penyadap berada di luar jaringan lokal.

Beberapa sistem *instant messaging* juga memiliki mekanisme koneksi dengan arsitektur peer-to-peer. Pada sistem dengan arsitektur peer-to-peer, *client* menghubungi *server* untuk mengetahui lokasi pihak lain yang akan dihubungi. Setelah mengetahui lokasinya, *client* akan menghubungi pihak lain secara langsung.

Semua informasi yang dibutuhkan agar program *client instant messaging* dapat menghubungi *server* telah diprekonfigurasi. Misalnya daftar alamat *server* yang harus dihubungi ketika akan memulai sesi percakapan. Setelah terhubung, *client instant*

messaging dapat bertukar pesan dengan *client* lainnya.



Gambar 2: Instant messaging dengan arsitektur peer-to-peer [1]

Banyak perusahaan dan organisasi yang memasang *firewall* pada jaringan internalnya. Tujuannya untuk membatasi akses hanya pada layanan tertentu yang dianggap penting, seperti layanan email, web, atau DNS. Oleh karena itu, penyedia layanan *instant messaging* biasanya mendesain agar program *client* layanan tersebut dapat melewati *firewall* dengan berbagai teknik, misalnya dengan membuat tunnel di atas layanan yang diizinkan oleh *firewall*.

Pada kebanyakan program *client instant messaging*, jika koneksi dengan *server* layanan gagal, maka *client* akan mencoba beberapa langkah lain agar dapat menghubungi *server*. Sebagai contoh, *client* akan mencoba mengontak *server* pada *port* yang umum seperti *port* 80 (HTTP). Jika *firewall* diset untuk mengizinkan paket melalui *port* ini, maka koneksi akan berhasil.

3.2 Protokol

Sebagian besar protokol yang digunakan pada layanan *instant messaging* bersifat tertutup (*proprietary*) dan tidak dipublikasikan. Namun, telah banyak pihak yang berusaha melakukan rekayasa balik (*reverse engineering*) terhadap protokol-protokol tersebut. Sehingga memudahkan pihak ketiga untuk mengembangkan program *client* untuk layanan tersebut.

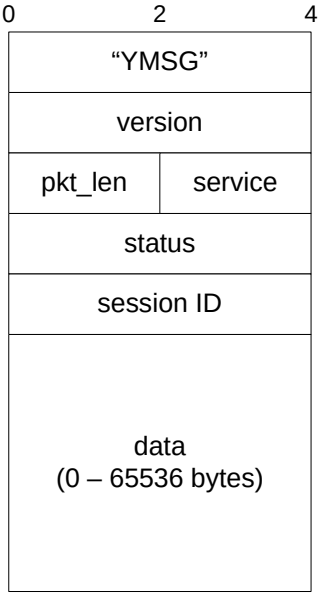
Proses rekayasa balik dilakukan oleh komunitas berdasarkan berbagai sumber serta hasil analisis terhadap paket data yang dikirimkan melalui jaringan serta analisis terhadap program *client* resmi yang dirilis oleh penyedia layanan.

Pada makalah ini, akan dipaparkan beberapa protokol yang digunakan pada layanan *instant messaging*, yaitu:

a. protokol YMSG yang digunakan pada layanan Yahoo Messenger
b. protokol MSNP yang digunakan pada layanan Windows Live Messenger
c. protokol Jabber yang digunakan pada layanan Google Talk

3.2.1 Protokol YMSG pada Yahoo Messenger

YMSG adalah protokol *proprietary* yang digunakan pada layanan *instant messaging* Yahoo Messenger. Dokumentasi protokol ini termasuk hasil rekayasa balik karena pihak Yahoo tidak pernah mempublikasikan spesifikasi protokolnya.



Gambar 3: Struktur paket YMSG [7]

Umumnya, YMSG berjalan pada layer aplikasi di atas TCP, namun pada kasus tertentu bisa juga berjalan di atas HTTP[2]. YMSG adalah

[2] Hyper-Text Transfer Protocol

protokol yang berbasis *byte*, sehingga sulit untuk dibaca dengan program editor teks biasa karena mengandung karakter yang tidak dapat ditampilkan. Struktur paket YMSG adalah seperti pada Gambar 3.

Keterangan mengenai *field-field* yang terdapat pada paket YMSG dapat dilihat pada Tabel 2.

Field "service" menandakan aksi yang dituju, misalnya untuk login atau untuk mengirim pesan. Sedangkan *field* "status" menandakan kode status atas *request* atau *response* yang dikirimkan sebelumnya.

Tabel 2: Daftar field pada paket YMSG [7]

Field	Keterangan
YMSG	Berisi karakter YMSG sebagai penanda protokol yang digunakan.
version	Versi protokol yang digunakan pada paket. Misalnya untuk versi 9 field ini berisi nilai 0x09000000.
pkt_len	Berisi panjang *field* data yang dikirimkan. Secara teoretis panjang *field* data dapat mencapai 65536 *bytes*. Namun biasanya tidak lebih dari 1000 *bytes*.
service	Menandakan service apa yang dituju, baik dalam *request* atau *response*.
status	Menandakan status, misalnya gagal atau berhasil.
session id	Sebagai penanda session apabila koneksi dilakukan melalui HTTP
data	Data yang dikirimkan. Terdiri atas pasangan *key* dan *value* yang masing-masing dipisahkan oleh string 0xC080

Tabel 3: Daftar status pada paket YMSG [7]

Status	Kode
YAHOO_STATUS_AVAILABLE	0x00000000
YAHOO_STATUS_BRB	0x00000001
YAHOO_STATUS_BUSY	0x00000002
YAHOO_STATUS_NOTATHOME	0x00000003
YAHOO_STATUS_NOTATDESK	0x00000004
YAHOO_STATUS_NOTINOFFICE	0x00000005
YAHOO_STATUS_ONPHONE	0x00000006
YAHOO_STATUS_ONVACATION	0x00000007
YAHOO_STATUS_OUTTOLUNCH	0x00000008
YAHOO_STATUS_STEPPEDOUT	0x00000009
YAHOO_STATUS_INVISIBLE	0x0000000C
YAHOO_STATUS_CUSTOM	0x00000063
YAHOO_STATUS_IDLE	0x000003E7
YAHOO_STATUS_OFFLINE	0x5a55aa56
YAHOO_STATUS_TYPING	0x00000016

Tabel 4: Daftar service pada paket YMSG[7]

Service	Kode
YAHOO_SERVICE_LOGON	0x01
YAHOO_SERVICE_LOGOFF	0x02
YAHOO_SERVICE_ISAWAY	0x03
YAHOO_SERVICE_ISBACK	0x04
YAHOO_SERVICE_IDLE	0x05
YAHOO_SERVICE_MESSAGE	0x06
YAHOO_SERVICE_IDACT	0x07
YAHOO_SERVICE_IDDEACT	0x08
YAHOO_SERVICE_MAILSTAT	0x09
YAHOO_SERVICE_USERSTAT	0x0a
YAHOO_SERVICE_NEWMAIL	0x0b
YAHOO_SERVICE_*CHAT*INVITE	0x0c
YAHOO_SERVICE_CALENDAR	0x0d
YAHOO_SERVICE_NEWPERSONALMAIL	0x0e
YAHOO_SERVICE_NEWCONTACT	0x0f
YAHOO_SERVICE_ADDIDENT	0x10
YAHOO_SERVICE_ADDIGNORE	0x11
YAHOO_SERVICE_PING	0x12
YAHOO_SERVICE_GROUPRENAME	0x13
YAHOO_SERVICE_SYSMESSAGE	0x14
YAHOO_SERVICE_PASSTHROUGH2	0x16
YAHOO_SERVICE_CONFINVITE	0x18
YAHOO_SERVICE_CONFLOGON	0x19

Service	Kode
YAHOO_SERVICE_CONFDECLINE	0x1a
YAHOO_SERVICE_CONFLOGOFF	0x1b
YAHOO_SERVICE_CONFADDINVITE	0x1c
YAHOO_SERVICE_CONFMSG	0x1d
YAHOO_SERVICE_*CHAT*LOGON	0x1e
YAHOO_SERVICE_*CHAT*LOGOFF	0x1f
YAHOO_SERVICE_*CHAT*MSG	0x20
YAHOO_SERVICE_GAMELOGON	0x28
YAHOO_SERVICE_GAMELOGOFF	0x29
YAHOO_SERVICE_GAMEMSG	0x2a
YAHOO_SERVICE_FILETRANSFER	0x46
YAHOO_SERVICE_*VOICECHAT*	0x4a
YAHOO_SERVICE_NOTIFY	0x4b
YAHOO_SERVICE_P2PFILEXFER	0x4d
YAHOO_SERVICE_PEERTOPEER	0x4f
YAHOO_SERVICE_AUTHRESP	0x54
YAHOO_SERVICE_LIST	0x55
YAHOO_SERVICE_AUTH	0x57
YAHOO_SERVICE_ADDBUDDY	0x83
YAHOO_SERVICE_REMBUDDY	0x84
YAHOO_SERVICE_IGNORECONTACT	0x85
YAHOO_SERVICE_REJECTCONTACT	0x86

Koneksi yang terjadi pada sebuah sesi Yahoo Messenger menggunakan TCP/IP. Secara default, program *client* akan membuat koneksi ke sebuah *server* Yahoo dengan alamat scs.yahoo.com dengan *port* TCP 5050. Namun, program *client* Yahoo Messenger memiliki kemampuan untuk menggunakan *port* lain atau bahkan membungkus paket dalam protokol HTTP seandainya komunikasi ke *port* tersebut diblok oleh *firewall*.

Sebuah sesi Yahoo Messenger memiliki 2 state, yaitu:

a. Authentication state
Sesi Yahoo Messenger dimulai dengan authentication state. *Client* mengirimkan username ke *server*. *Server* kemudian akan merespon dengan *challenge string*. *Client* akan membalas *challenge string* dengan dua *response string*. Jika autentikasi berhasil, sesi akan berpindah ke *messaging state*. Jika tidak, *client* akan menerima pesan *error*.

b. Messaging state
Setelah proses autentikasi berhasil, sesi berpindah ke *messaging state*. *Server* akan mengirimkan *buddy list*, *ignore list*, *identity list*, dan *cookie list*. Kemudian *server* akan mengirimkan daftar kontak yang *online* beserta statusnya masing-masing. Pada saat ini, pesan-pesan offline akan dikirimkan kepada *client*.
Pada state ini, *client* dapat mengirim dan menerima pesan, bergabung dengan *conference*, mengirim dan menerima file, mengubah status, dan aksi lainnya. State ini berakhir ketika pengguna offline dengan mengirimkan paket LOGOFF.

Ketika pengguna mengirim pesan, maka program *client* akan mengirimkan paket dengan isi *field* service 0x06 dan isi *field* status 0x00000000. *Field* data akan diisi dengan serangkaian key dan value berikut:

a. *key* 0 dengan *value* berupa Yahoo ID pengirim
b. *key* 1 dengan *value* berupa Yahoo ID aktif pengirim
c. *key* 5 dengan *value* berupa Yahoo ID penerima pesan
d. *key* 14 dengan *value* berupa isi pesan yang akan dikirimkan

Layanan Yahoo Messenger tidak menyediakan fasilitas enkripsi pesan, sehingga pesan yang dikirimkan antara *client* dan *server* berupa plainteks.

3.2.2 Protokol MSNP pada Windows Live Messenger

MSNP (Mobile Status Notification Protocol) adalah protokol *proprietary* yang digunakan oleh Microsoft pada layanan Windows Live Messenger. Saat ini MSNP sudah mencapai versi 14. Microsoft pernah mempublikasikan versi 2 protokol ini dalam sebuah internet draft, namun versi selanjutnya tidak pernah dipublikasikan lagi. Sehingga dokumentasi protokol ini termasuk hasil rekayasa balik [8].

Protokol MSNP adalah protokol yang berbasis teks dan *human-readable* (dapat dibaca). Protokol ini telah mengalami sejumlah revisi. Masing-masing versi disebut dengan nama MSN#, misalnya MSNP8, MSNP9, atau MSNP14.

Protokol MSNP terdiri atas serangkaian perintah yang ditransmisikan di antara *client* dan *server*. Sebuah sesi Windows Live Messenger melibatkan koneksi ke sebuah *Notification Server* (NS) yang menyediakan layanan *presence* dan koneksi ke *Switchboard Server* (SB) yang menyediakan layanan messaging.

Koneksi ke *notification server* merupakan dasar sebuah sesi Windows Live Messenger. Fungsi utama NS adalah menangani informasi *presence* pengguna dan daftar kontak yang dimiliki pengguna. Jika koneksi ke NS terputus, maka pengguna akan terlihat *offline* oleh semua temannya.

Notification *server* juga menangani beberapa hal, seperti informasi mengenai email baru yang masuk ke *mailbox* dan pembuatan sesi *switchboard*. Ketika pengguna bergabung dengan sebuah sesi *switchboard*, koneksi ke NS akan tetap terbuka.

Switchboard Server menangani sesi messaging antar pengguna. Setiap sesi percakapan pada Windows Live Messenger melibatkan koneksi ke sebuah sesi *switchboard*. *Switchboard* berlaku sebagai *proxy* antara pengguna dengan kontak yang diajak berkomunikasi. Sebuah sesi *switchboard* dapat diikuti oleh beberapa orang.

Semua koneksi pada sesi MSNP dilakukan dengan TCP/IP dengan *port* default TCP 1863. Namun, MSNP dapat menggunakan *port* yang lain, misalnya ketika *port* tersebut diblok oleh *firewall*. Sesi MSNP dapat dibungkus dengan HTTP, sehingga dapat memanfaatkan *proxy server*.

```
MSG alice@passport.com Alice
143\r\n
MIME-Version: 1.0\r\n
Content-Type: text/plain;
charset=UTF-8\r\n
X-MMS-IM-Format:
FN=Lucida%20Sans%20Unicode;
EF=B; CO=ff0000; CS=0; PF=22\r\n
\r\n
```

Gambar 4: Contoh isi pesan pada protokol MSNP

Data dikirim antara *client* dan *server* dalam bentuk command. Command dilambangkan dengan 3 huruf kapital. Ada 5 jenis command yang dikenal pada MSNP, yaitu:

a. *Normal command*
   Perintah biasa yang memiliki parameter transaction ID dan selalu diakhiri dengan newline.
b. *Payload command*
   Perintah yang dapat terdiri dari beberapa baris.
c. *Error command*
   Berbeda dengan jenis perintah yang lain, command code pada error command berupa rangkaian 3 buah angka.
d. *Asynchronous command*
   Perintah yang dikirim oleh *server* tanpa diminta oleh *client* secara eksplisit.
e. *Special command*
   Contoh special command adalah PNG command & SYN response yang tidak memiliki transaction ID.

Ketika pengguna mengirim pesan, maka program *client* akan mengirimkan *payload command* MSG dengan *payload* berupa isi pesan yang dikirimkan. Contoh isi pesan dapat dilihat pada Gambar 4.

Layanan Windows Live Messenger tidak menyediakan fasilitas enkripsi, sehingga pesan yang dikirim antara *client* dan *server* berupa plainteks.

3.2.3 Protokol Jabber pada Google Talk

Jabber adalah protokol yang bersifat terbuka dan telah distandardisasi. Protokol ini berbasis XML, sehingga protokolnya bersifat human readable. Protokol ini dikenal juga dengan nama XMPP (*Extended Messaging and Presence Protocol*). Spesifikasi Jabber tersedia secara publik dan didefinisikan pada RFC (Request For Comment) 3920 [9].

Layanan Google Talk menggunakan Jabber sebagai protokol *instant messaging* dan beberapa protokol lain buatannya sendiri untuk mendukung *voice chat*, yaitu Jingle.

Karena Google Talk menggunakan Jabber, maka Google Talk mewarisi kelebihan-kelebihan protokol Jabber. Antara lain dapat menggunakan program *client* berbasis Jabber yang banyak tersedia, baik gratis atau komersial. Selain itu, protokol Jabber bersifat terdistribusi dan terdesentralisasi, sehingga pengguna Google Talk dapat menghubungi pengguna layanan lain yang sama-sama berbasis Jabber.

Pada sebuah sesi komunikasi dengan protokol Jabber, koneksi dilakukan melalui TCP/IP

dengan *port* default TCP 5222. Secara default, koneksi yang terjadi tidak dienkripsi.

Jabber menyediakan enkripsi dengan TLS (*Transport Layer Security*) melalui *port* 5223. Layanan Google Talk sendiri hanya menerima koneksi yang terenkripsi dengan TLS, sehingga lalu lintas data antara *client* dan *server* dalam bentuk terenkripsi.

Transport Layer Security (TLS) adalah protokol kriptografi yang menyediakan saluran yang aman bagi komunikasi via email, *instant messaging*, dan komunikasi lain. Saat ini TLS sudah mencapai versi 1.1. TLS adalah penerus dari protokol SSL (Secure Sockets Layer).

TLS menyediakan autentikasi dari ujung ke ujung serta privasi komunikasi dengan menggunakan kriptografi. Protokol ini didesain untuk mencegah penyadapan maupun pengubahan pesan.

4 Analisis Komunikasi Data

Pada makalah ini, akan diamati lalu lintas pesan antara *client* dan *server instant messaging*. Data yang diamati hanya yang berupa pesan (message). Layanan *instant messaging* yang akan dianalisis hanya dibatasi pada layanan berikut:

a. Yahoo Messenger
b. Windows Live Messenger
c. Google Talk

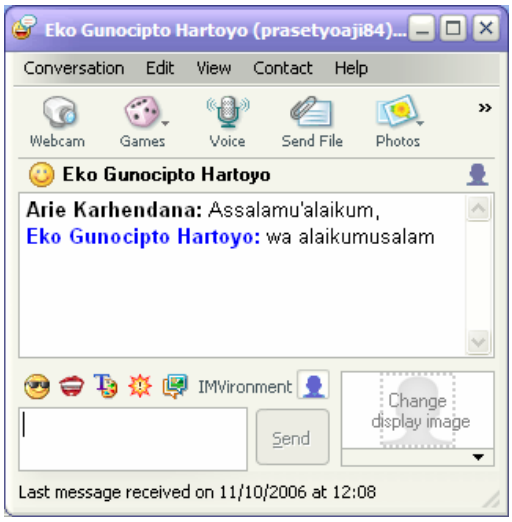
Proses pengamatan lalu lintas data pada jaringan akan dilakukan dengan teknik *sniffing*. *Sniffing* adalah teknik yang mencoba merekan semua aktivitas lalu lintas data pada jaringan dengan bantuan kakas *sniffer*. *Sniffer* adalah perangkat yang dapat merekam (*capture*) semua aktivitas lalu lintas data pada jaringan. Namun *sniffer* hanya dapat merekam trafik yang sampai ke komputer kita. Untuk keperluan ini, kakas *sniffer* yang digunakan adalah Ethereal.

4.1 Analisis Pesan pada Yahoo Messenger

Pada pengujian ini, dilakukan percobaan dengan menggunakan program *client* Yahoo Messenger versi 7.

Sniffer diset untuk merekam semua aliran data pada jaringan dengan protokol TCP, baik paket yang bertujuan ke *port* 5050 (dari *client* ke *server*), maupun paket yang berasal dari *port* 5050 (dari *server* ke *client*).

Pada sesi percakapan ini, dikirimkan sebuah pesan singkat yang kemudian dibalas oleh pesan singkat pula oleh rekan pengguna. Pesan yang dikirimkan adalah seperti pada Gambar 5.



Gambar 5: Contoh percakapan melalui Yahoo Messenger

Hasil capture paket pada jaringan yang berisi pesan pertama dapat dilihat pada Gambar 6. Pesan ini dikirim oleh *client* ke *server*. Sedangkan hasil capture paket yang berisi pesan kedua dapat dilihat pada Gambar 7. Pesan ini dikirim oleh *server* ke *client*.

Dapat dilihat bahwa pesan yang dikirimkan dapat terbaca dengan jelas karena berupa plainteks.

Dari hasil capture kedua pesan ini, dapat disimpulkan bahwa komunikasi yang terjadi antara *client* dan *server* pada sesi percakapan dengan Yahoo Messenger dapat disadap dengan mudah karena isinya berupa plainteks yang tidak terenkripsi.

4.2 Analisis Pesan pada Windows Live Messenger

Pada pengujian ini, dilakukan percobaan dengan menggunakan program *client* Windows Live Messenger versi 8.

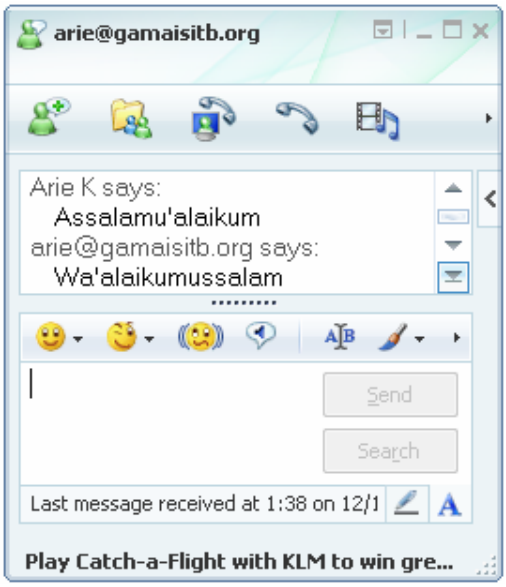
Kakas *sniffer* diset untuk merekam semua aliran data pada jaringan dengan protokol TCP, baik paket yang bertujuan ke *port* 1863 (dari *client* ke *server*), maupun paket yang berasal dari *port* 1863 (dari *server* ke *client*).

```
59 4D 53 47 00 0C 00 00-00 64 00 06 5A 55 AA 56  YMSG.....d..ZU.V
9A 80 E2 68 31 C0 80 61-72 69 65 5F 6B 61 72 68  ...h1..arie_karh
65 6E 64 61 6E 61 C0 80-35 C0 80 70 72 61 73 65  endana..5..prase
74 79 6F 61 6A 69 38 34-C0 80 31 34 C0 80 41 73  tyoaji84..14..As
73 61 6C 61 6D 75 27 61-6C 61 69 6B 75 6D 2C C0  salamu'alaikum,.
80 39 37 C0 80 31 C0 80-36 33 C0 80 3B 30 C0 80  .97..1..63..;0..
36 34 C0 80 30 C0 80 31-30 30 32 C0 80 31 C0 80  64..0..1002..1..
32 30 36 C0 80 30 C0 80                          206..0..
```

Gambar 6: Hasil capture pesan 1 pada Yahoo Messenger

```
59 4D 53 47 00 00 00 00-00 79 00 06 00 00 00 01  YMSG.....y......
9A 80 E2 68 35 C0 80 61-72 69 65 5F 6B 61 72 68  ...h5..arie_karh
65 6E 64 61 6E 61 C0 80-34 C0 80 70 72 61 73 65  endana..4..prase
74 79 6F 61 6A 69 38 34-C0 80 32 35 32 C0 80 43  tyoaji84..252..C
5A 2F 6A 34 63 49 54 30-2B 66 73 52 35 58 73 55  Z/j4cIT0+fsR5XsU
46 49 7A 49 6A 6B 5A 65-34 78 46 53 51 3D 3D C0  FIzIjkZe4xFSQ==.
80 39 37 C0 80 31 C0 80-31 34 C0 80 77 61 20 61  .97..1..14..wa a
6C 61 69 6B 75 6D 75 73-61 6C 61 6D C0 80 36 33  laikumusalam..63
C0 80 3B 30 C0 80 36 34-C0 80 30 C0 80           ..;0..64..0..
```

Gambar 7: Hasil capture pesan 2 pada Yahoo Messenger



Gambar 8: Contoh percakapan melalui Windows Live Messenger

Pada sesi percakapan ini, dikirimkan sebuah pesan singkat yang kemudian dibalas oleh pesan singkat pula oleh rekan pengguna. Pesan yang dikirimkan adalah seperti pada Gambar 8.

Hasil capture paket pada jaringan yang berisi pesan pertama dapat dilihat pada Gambar 9. Pesan ini dikirim oleh *client* ke *server*. Sedangkan hasil capture paket yang berisi pesan kedua dapat dilihat pada Gambar 10. Pesan ini dikirim oleh *server* ke *client*.

Dapat dilihat bahwa pesan yang dikirimkan dapat terbaca dengan jelas karena berupa plainteks.

Dari hasil capture kedua pesan ini, dapat disimpulkan bahwa komunikasi yang terjadi antara *client* dan *server* pada sesi percakapan dengan Windows Live Messenger dapat disadap dengan mudah karena isinya berupa plainteks yang tidak terenkripsi.

```
4D 53 47 20 35 20 4E 20-31 33 39 0D 0A 4D 49 4D  MSG 5 N 139..MIM
45 2D 56 65 72 73 69 6F-6E 3A 20 31 2E 30 0D 0A  E-Version: 1.0..
43 6F 6E 74 65 6E 74 2D-54 79 70 65 3A 20 74 65  Content-Type: te
78 74 2F 70 6C 61 69 6E-3B 20 63 68 61 72 73 65  xt/plain; charse
74 3D 55 54 46 2D 38 0D-0A 58 2D 4D 4D 53 2D 49  t=UTF-8..X-MMS-I
4D 2D 46 6F 72 6D 61 74-3A 20 46 4E 3D 4D 53 25  M-Format: FN=MS%
32 30 53 68 65 6C 6C 25-32 30 44 6C 67 3B 20 45  20Shell%20Dlg; E
46 3D 3B 20 43 4F 3D 30-3B 20 43 53 3D 30 3B 20  F=; CO=0; CS=0;
50 46 3D 30 0D 0A 0D 0A-41 73 73 61 6C 61 6D 75  PF=0....Assalamu
27 61 6C 61 69 6B 75 6D                          'alaikum
```

Gambar 9: Hasil capture pesan 1 pada Windows Live Messenger

```
4D 53 47 20 61 72 69 65-40 67 61 6D 61 69 73 69  MSG arie@gamaisi
74 62 2E 6F 72 67 20 61-72 69 65 40 67 61 6D 61  tb.org arie@gama
69 73 69 74 62 2E 6F 72-67 20 31 36 31 0D 0A 4D  isitb.org 161..M
49 4D 45 2D 56 65 72 73-69 6F 6E 3A 20 31 2E 30  IME-Version: 1.0
0D 0A 43 6F 6E 74 65 6E-74 2D 54 79 70 65 3A 20  ..Content-Type:
74 65 78 74 2F 70 6C 61-69 6E 3B 20 63 68 61 72  text/plain; char
73 65 74 3D 55 54 46 2D-38 0D 0A 55 73 65 72 2D  set=UTF-8..User-
41 67 65 6E 74 3A 20 56-65 72 69 43 68 61 74 2F  Agent: VeriChat/
31 2E 35 0D 0A 58 2D 4D-4D 53 2D 49 4D 2D 46 6F  1.5..X-MMS-IM-Fo
72 6D 61 74 3A 20 46 4E-3D 4D 53 25 32 30 53 61  rmat: FN=MS%20Sa
6E 73 25 32 30 53 65 72-69 66 3B 20 45 46 3D 3B  ns%20Serif; EF=;
20 43 4F 3D 30 3B 20 50-46 3D 30 0D 0A 0D 0A 57   CO=0; PF=0....W
61 27 61 6C 61 69 6B 75-6D 75 73 73 61 6C 61 6D  a'alaikumussalam
```

Gambar 10: Hasil capture pesan 2 pada Windows Live Messenger

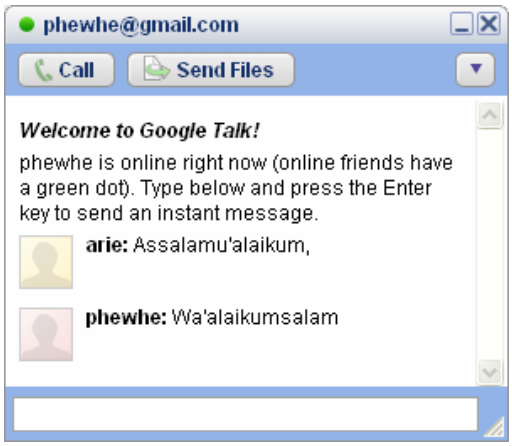
4.3 Analisis Pesan pada Google Talk

Pada pengujian ini, dilakukan percobaan dengan menggunakan program *client* Google Talk.

Kakas *sniffer* diset untuk merekam semua aliran data pada jaringan dengan protokol TCP, baik paket yang bertujuan ke *port* 5222 (dari *client* ke *server*), maupun paket yang berasal dari *port* 5222 (dari *server* ke *client*).

Pada sesi percakapan ini, dikirimkan sebuah pesan singkat yang kemudian dibalas oleh pesan singkat pula oleh rekan pengguna. Pesan yang dikirimkan adalah seperti pada Gambar 11.

Sesi percakapan dengan menggunakan Google Talk lebih sulit di-capture karena lalu lintas data antara *client* dan *server* berupa pesan terenkripsi. Salah satu potongan stream selama komunikasi dapat dilihat pada Gambar 12.



Gambar 11: Contoh percakapan melalui Google Talk

```
17 03 01 01 8D 8B 87 5A-5C 11 ED F7 56 BE 53 E4  .......Z\...V.S.
39 75 5B 69 E6 2F CB 5E-99 2F 8D 27 17 55 C5 97  9u[i./.^./.'.U..
55 FE 8C D6 7D E7 18 EB-2A FC FD 62 03 EC 88 62  U...}...*..b...b
8B E9 C4 5F C3 A5 42 19-6C 29 30 80 2A 35 61 28  ..._..B.l)0.*5a(
F6 30 08 77 9E D3 A8 43-FA EC 48 6C A8 AD C7 EA  .0.w...C..Hl....
14 5D 0B 59 C8 F7 31 A6-28 F5 52 F5 E0 98 B1 8B  .].Y..1.(.R.....
91 6E 5B 9B 25 30 97 76-15 66 03 40 BB 39 93 2E  .n[.%0.v.f.@.9..
EC 96 20 28 CF 08 5D 54-F1 E9 AF E5 45 EC 1B E1  .. (..]T....E...
6D C3 95 1B 01 06 F4 4B-14 B4 CA 1E A1 86 58 98  m......K......X.
4B D0 6B 95 BF 02 1E 95-02 7B 3E C2 5C F7 17 46  K.k......{>.\..F
A7 E9 95 6E 54 74 06 1D-0C 90 D8 41 8B 58 85 E4  ...nTt.....A.X..
6A 62 8C 2A 42 F8 CC D7-D0 C3 81 64 D6 C8 8C 05  jb.*B......d....
FC 8C 06 1D 1C 05 8E A6-7A 4F 77 2C B8 95 2C 91  ........zOw,..,.
A2 F3 BC B6 47 3F CE 99-B2 55 90 0B 33 0B 05 0D  ....G?...U..3...
50 6D B4 08 10 8A B9 AC-3D 1C 03 78 DE 0E E3 B6  Pm......=..x....
82 1B EE 50 E0 8E F3 D1-FB 85 12 72 00 FB 10 D3  ...P.......r....
2D B5 07 04 55 16 7B C1-30 81 63 FF 15 7F 7B 5F  -...U.{.0.c...{_
99 75 B3 14 68 EF 0C CF-7E 17 DE F2 C9 CB 48 A4  .u..h...~.....H.
BA 93 85 E4 F0 D8 38 BD-97 4A DC CA 0F FA 4F A4  ......8..J....O.
78 FF 79 EB 4B 8B 72 61-AB D8 C6 CA D1 40 99 9E  x.y.K.ra.....@..
E2 50 7F 55 D9 8B 5B 8E-A4 5C A4 A2 D8 CF A3 92  .P.U..[..\......
F9 71 DD 0C 31 DA 1F EC-15 55 90 87 0C 9A 2B E9  .q..1....U....+.
9D 8A 11 09 5D A9 F8 FA-24 A6 18 1F F4 B5 40 E4  ....]...$.....@.
0F FC C4 FB 40 37 83 6C-0C A2 59 4F 0E 09 F1 0B  ....@7.l..YO....
5C 46 33 67 C7 8A 15 F5-10 AD 29 4A E2 84 0A 4A  \F3g......)J...J
06 E6                                            ..
```

Gambar 12: Salah satu potongan pesan dalam sesi percakapan Google Talk

Pada saat awal koneksi, program *client* akan menegosiasikan metode enkripsi yang akan digunakan. Sehingga pada awal koneksi, komunikasi yang terjadi antara *client* dan *server* masih dalam bentuk plainteks, seperti terlihat pada Gambar 13. Setelah negosiasi berhasil, maka komunikasi akan dilakukan dengan pesan yang terenkripsi.

```
<stream:stream to="gmail.com" xml:lang="en" version="1.0"
xmlns:stream="http://etherx.jabber.org/streams" xmlns="jabber:client">

<?xml version="1.0" encoding="UTF-8"?><stream:stream from="gmail.com"
id="X79871EE75FB937B4" version="1.0"
xmlns:stream="http://etherx.jabber.org/streams"
xmlns="jabber:client"><stream:features><starttls
xmlns="urn:ietf:params:xml:ns:xmpp-tls"/><mechanisms
xmlns="urn:ietf:params:xml:ns:xmpp-sasl"><mechanism>X-GOOGLE-
TOKEN</mechanism></mechanisms></stream:features>

<starttls xmlns="urn:ietf:params:xml:ns:xmpp-tls"/>

<proceed xmlns="urn:ietf:params:xml:ns:xmpp-tls"/>
```

Gambar 13: Proses negosiasi enkripsi antara client dan server pada Google Talk

4.4 Hasil Pengujian

Dari hasil pengujian tersebut, dapat disimpulkan beberapa hasil berikut:

a. Pada Yahoo Messenger, pesan yang dikirimkan antara *client* dan *server* berupa plainteks, sehingga dapat disadap dalam perjalanan.
b. Demikian pula pada Windows Live Messenger, pesan yang dikirimkan antara *client* dan *server* juga berupa plainteks.
c. Layanan Google Talk relatif lebih aman karena pesan yang dikirimkan antara *client* dan *server* sudah dalam bentuk terenkripsi.

5 Kesimpulan

Tingkat penggunaan *instant messaging* diperkirakan akan semakin tinggi pada masa yang akan datang, baik untuk pengguna rumah, perusahaan, maupun pengguna yang mobile dengan penggunaan teknologi nirkabel.

Instant messaging diperkirakan akan menjadi teknologi yang memegang peranan penting, seperti email dan telepon. Idealnya, penggunaan *instant messaging* dapat mendekatkan hubungan antara personal.

Namun, peningkatan ketergantungan terhadap layanan ini menimbulkan akibat lain. Sehingga kerentanan sekecil apa pun pada layanan *instant messaging* dapat membawa dampak negatif yang sangat besar.

Dari hasil pengujian ini, dapat disimpulkan bahwa komunikasi yang terjadi antara *client* dan *server* pada sesi percakapan dengan Google Talk sulit untuk disadap dengan teknik *sniffing* karena isi pesannya terenkripsi.

Dari hasil paparan makalah ini, dapat disimpulkan bahwa:

a. Kebanyakan layanan *instant messaging* saat ini didesain bukan berdasarkan aspek keamanan sebagai pertimbangan utama, melainkan aspek skalabilitas untuk menunjang jumlah pengguna yang begitu besar
b. Penggunaan *instant messaging* memang memudahkan dan sampai batas tertentu dapat meningkatkan produktivitas. Namun, terlepas dari itu, penggunaan *instant messaging* yang tidak tepat merupakan ancaman yang sangat besar terhadap keamanan data, terutama bagi organisasi dan korporat.
c. Layanan *instant messaging* seperti Yahoo Messenger dan Windows Live Messenger, tidak aman untuk digunakan karena data yang ditransmisikan menggunakan kedua layanan tersebut dapat dibaca oleh penyadap. Sedangkan layanan Google Talk relatif lebih aman karena komunikasi yang terjadi menggunakan enkripsi.

Pihak-pihak yang sangat tergantung kepada layanan *instant messaging* namun menginginkan tingkat keamanan yang lebih tinggi diharapkan menggunakan solusi *instant messaging* yang aman, antara lain:

a. Memasang sistem *instant messaging* privat untuk internal organisasi sehingga kecil kemungkinan komunikasi dapat disadap oleh pihak luar.
b. Menggunakan perangkat lunak pihak ketiga yang dapat melakukan enkripsi terhadap layanan *instant messaging* publik yang tidak aman. Kakas seperti ini dapat banyak ditemukan di internet, baik dengan lisensi gratis maupun komersial.

Daftar Pustaka

[1] Symantec Enterprise Security. (2002). *Securing Instant Messaging*.

[2] Viewz. *Instant Messaging Guide*. *http://www.viewz.com/features/imguide.shtml*. Tanggal akses: 10 Oktober 2006.

[3] Wikipedia. *http://en.wikipedia.org*. Tanggal akses: 10 Oktober 2006.

[4] How Stuffs Work. *Hows Instant Messaging Works. http://www.howstuffworks.com/instant-messaging.htm*. Tanggal akses: 10 Oktober 2006.

[5] Symantec Security Response. (2006). *Top Five Instant Messaging Security Risks for 2006*.

[6] Nielsen/Netratings. *IM Market Share*. *http://www.bigblueball.com/forums/general-im-news/34413-im-market-share.html*. Tanggal akses: 10 Oktober 2006.

[7] *Yahoo Messenger Protocol Unofficial Documentation. http://libyahoo2.sourceforge.net*. Tanggal akses: 10 Oktober 2006.

[8] *MSN Protocol Version 8 Documentation. http://msnpiki.msnfanatic.com*. Tanggal akses: 10 Oktober 2006.

[9] Jabber Software Foundation. *http://www.jabber.org*. Tanggal akses: 10 Oktober 2006.