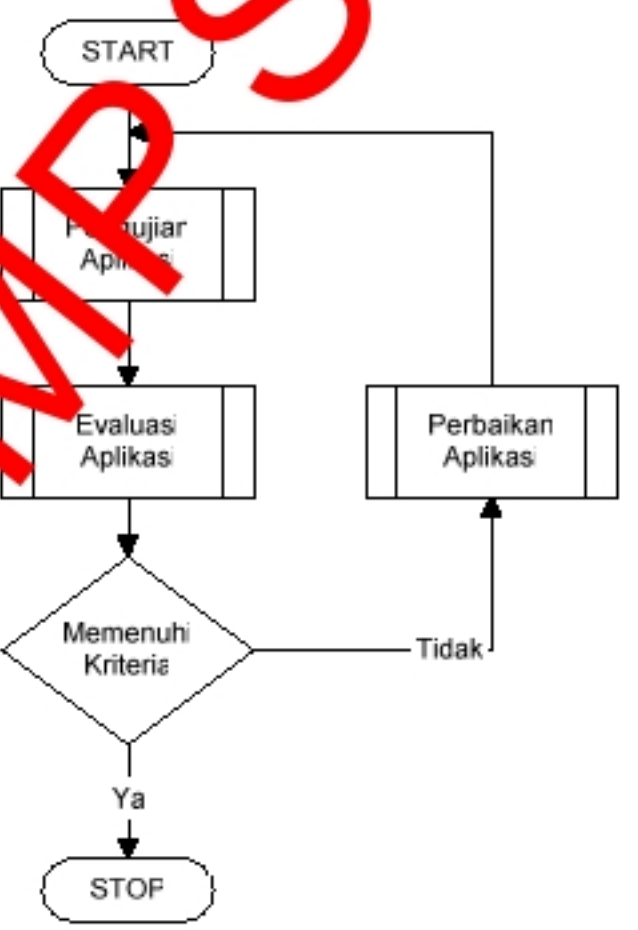
BAB IV

PENGUJIAN DAN EVALUASI SISTEM

4.1 Prosedur Pengujian

Untuk mengetahui kriteria aplikasi yang dibuat sudah memenuhi target yang ingin dicapai maka diperlukan adanya suatu pengujian dan evaluasi terhadap aplikasi yang dibuat. Pada proses pengujian dan evaluasi yang dilakukan pada program aplikasi sangat dimungkinkan adanya perbaikan – perbaikan pada program aplikasi. Apabila dalam hasil pengujian tidak dicapai hasil yang diinginkan, sehingga perbaikan – perbaikan akan terus dilakukan sehingga hasil yang diinginkan dapat tercapai. Secara umum, algoritma pengujian dan evaluasi sistem dapat dilihat pada gambar 4.1.



Gambar 4.1 Algoritma Pengujian dan Evaluasi aplikasi

STIKOM SURABAYA

Pada prosedur pengujian dilakukan secara bertahap, mulai dari bagian – bagian atau unit – unit kecil pada aplikasi sampai dengan yang terakhir adalah pengujian aplikasi secara keseluruhan dalam artian aplikasi sudah terintegrasi penuh antar semua bagian atau unit.

Pengujian dilakukan dengan langkah-langkah sebagai berikut :

1. Menjalankan program aplikasi server chat.
2. Menjalankan program aplikasi desktop client maupun website dan melakukan koneksi chatting ke server.
3. Client melakukan pengiriman pesan teks.
4. Client melakukan pengiriman file.

Prosedur evaluasi pengujian dilakukan secara bertahap, disesuaikan dengan langkah-lankah prosedur pengujian, hasil dari pengujian aplikasi akan dijadikan acuan sebagai bahan evaluasi.

Evaluasi dilakukan dengan kriteria-kriteria sebagai berikut :

1. Apakah server chat dapat menerima koneksi dari client.
2. Apakah server dapat menerima dan melakukan pengiriman pesan teks.
3. Apakah server mampu melakukan enkripsi dan dekripsi pesan teks dari client.
4. Apakah client dapat menggunakan fitur pengiriman file.

4.1.1 Mengaktifkan Servis Apache Tomcat

Untuk mengaktifkan Tomcat digunakan perintah " startup.bat " . Perintah ini hanya dapat dilakukan oleh user – user yang memiliki hak akses setingkat root / administrator.

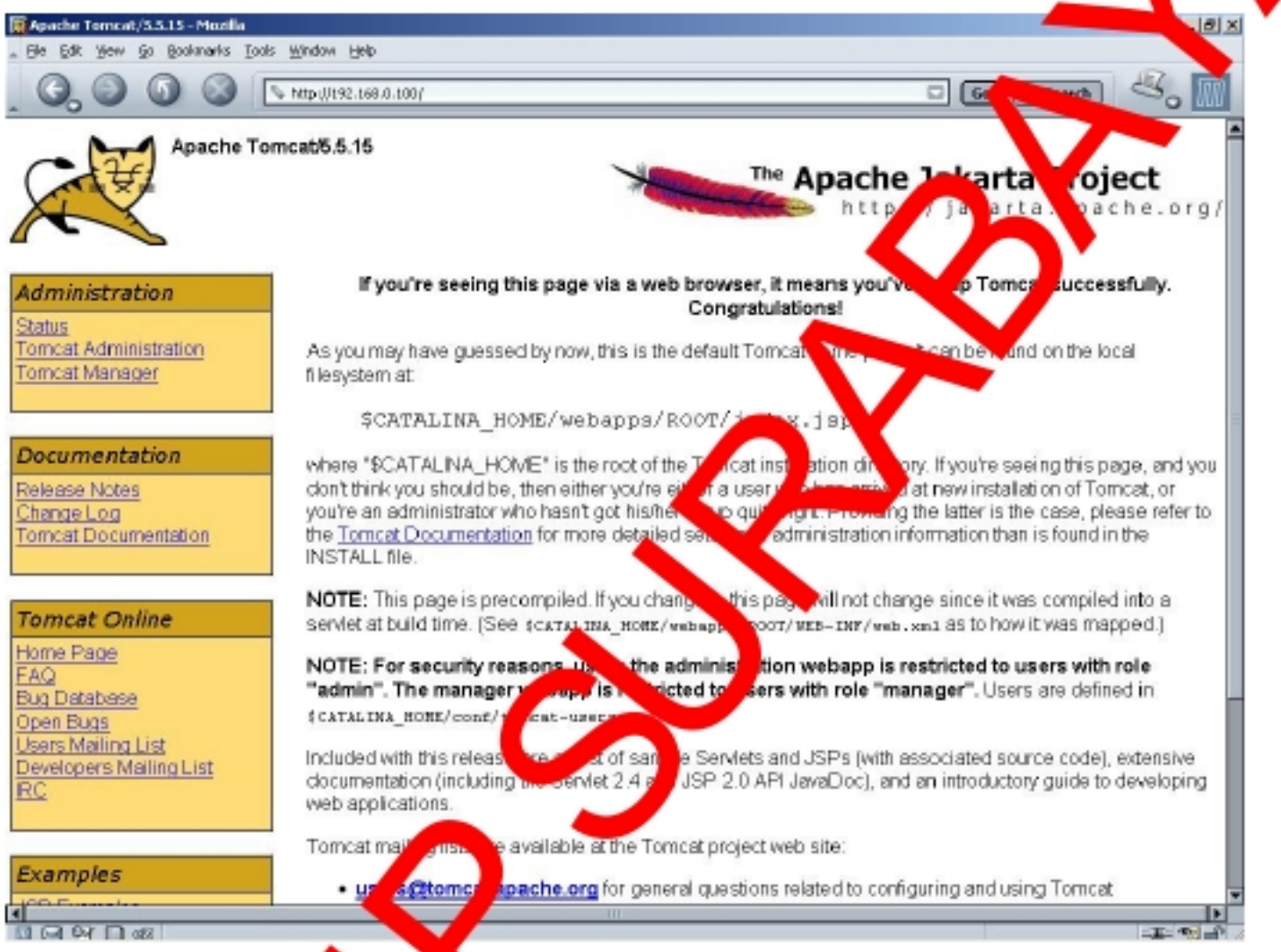
```
C:\Program Files\Apache Software Foundation\Tomcat 5.5\bin>tomcatw.exe
```

atau

```
C:\Program Files\Apache Software Foundation\Tomcat 5.5\bin>startup.bat
```

STIKOM SURABAYA

Untuk mengetahui apakah servis dari apache tomcat sudah berjalan dengan baik atau tidak, buka aplikasi browser dan gunakan alamat URL localhost:8080.



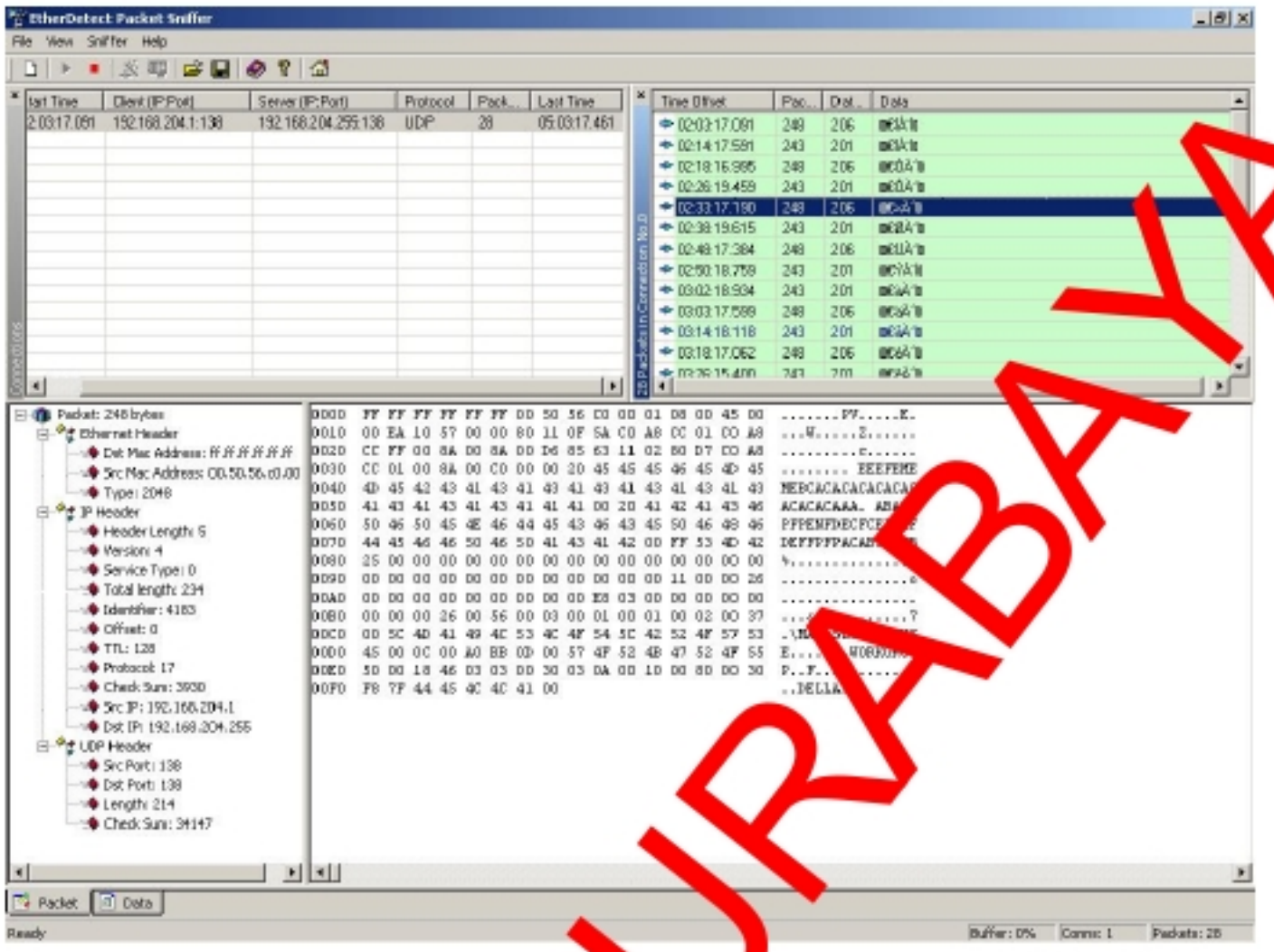
Gambar 4.2 Tomcat Web Server

4.1.2 Mengaktifkan Ether Detect Packet Sniffer(Aplikasi Sniffer)

Untuk mengaktifkan Ether Detect pada system operasi Microsoft Windows dapat dilakukan melalui shortcut aplikasi yang ada di menu Start >> Programs >> Ether Detects atau melalui shotcut aplikasi yang berada di desktop.

Untuk menjalankan fungsi sniffing, melalui menu Sniffer >> Start. Dibawah ini adalah tampilan dari aplikasi Ether Detect.

STIKOM SURABAYA

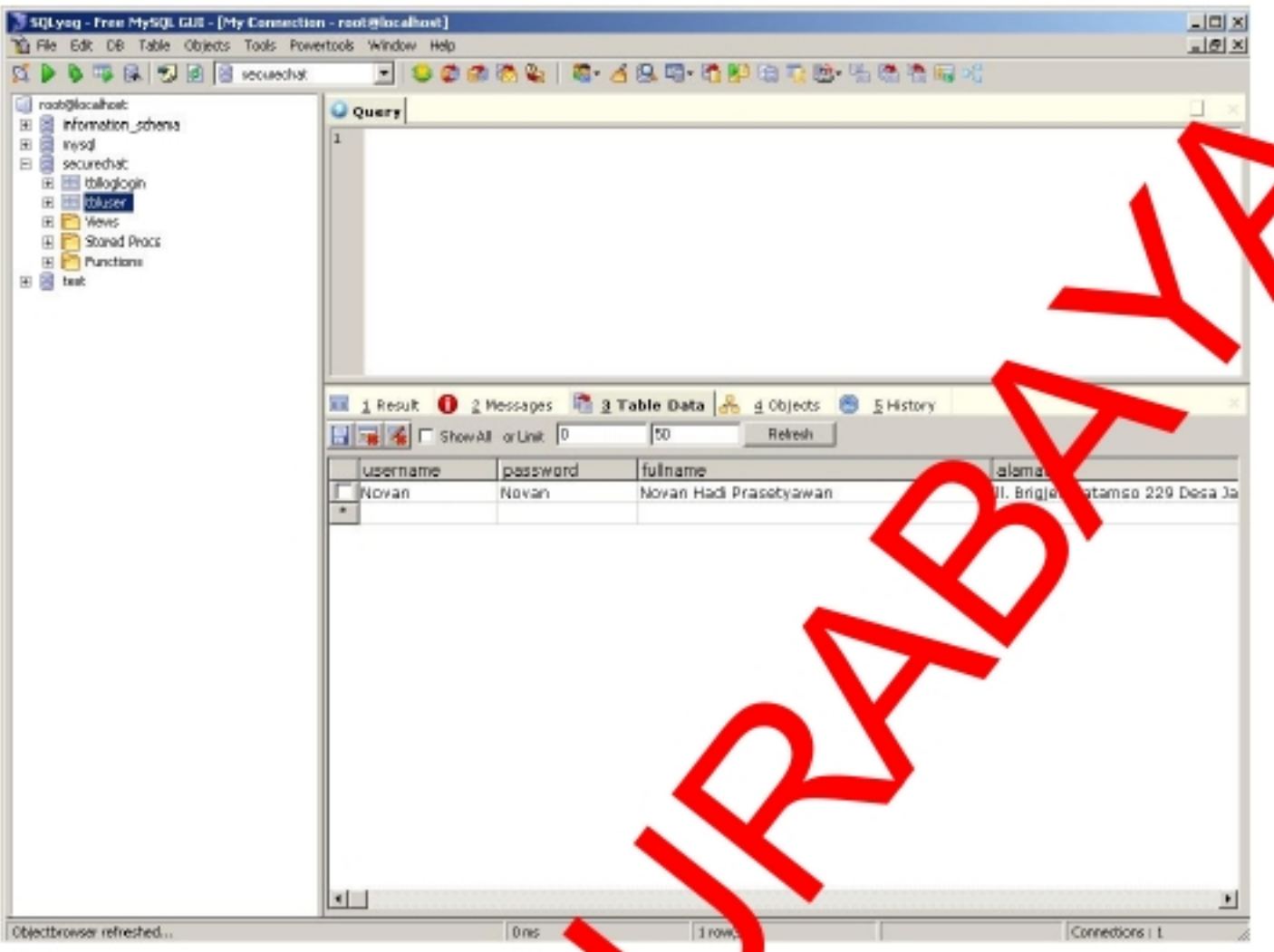


Gambar 4.3 EtherDetect Packet Sniffer

4.1.3 Mengaktifkan MySQL 5.0 Database Server

Untuk mengaktifkan Database Server MySQL dapat dilakukan melalui Control Panel >> Administrative Tools >> Service, aktifkan service dengan nama service mysql

SURABAYA
STIKOMPS



Gambar 4.4 SQL Yog (MySQL interface)

4.1.4 Mengaktifkan Aplikasi Chatting

A. ChatServer.java

Aplikasi ini berfungsi sebagai server chat yang bertugas untuk menghandle koneksi dari klien yang ingin melakukan chatting. Untuk mengaktifkan dan menjalankan aplikasi.

```
D:\SKRIPSI-NOVAN\SKRIPSI-NOVAN\Program TA + Fitur
File\ChatServer\java -jar ChatServer.jar
```

B. ChatClient.java

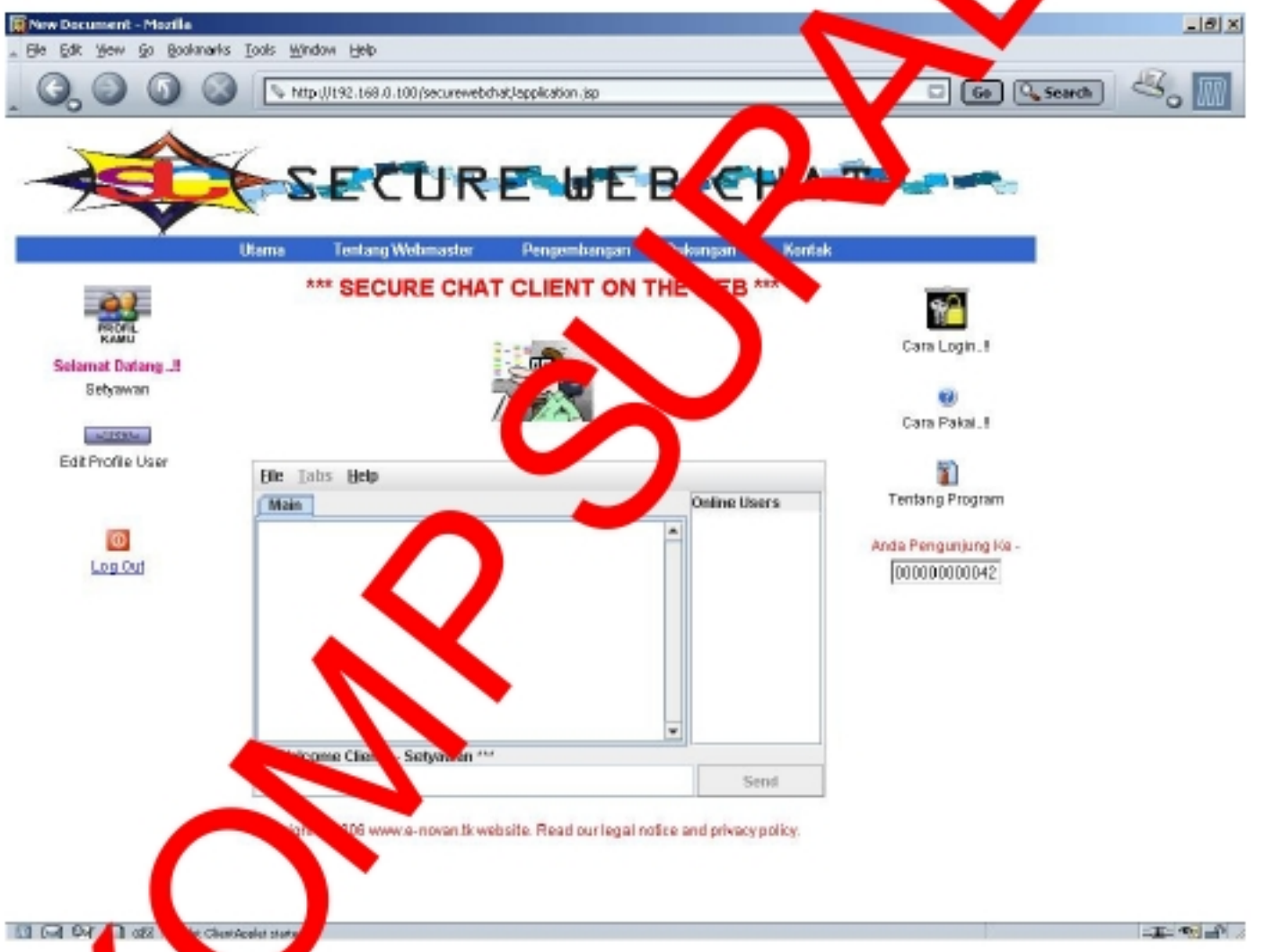
Aplikasi ini berfungsi untuk melakukan koneksi ke server chatting. Aplikasi ini hanya digunakan untuk user yang melakukan koneksi melalui aplikasi desktop tidak melalui web browser. Untuk mengaktifkan dan menjalankan aplikasi.

SURABAYA
STIKOMPS

```
D:\SKRIPSI-NOVAN\SKRIPSI-NOVAN\Program TA + Fitur
File\ChatClient\java -jar ChatClient.jar
```

C. ClientApplet.java

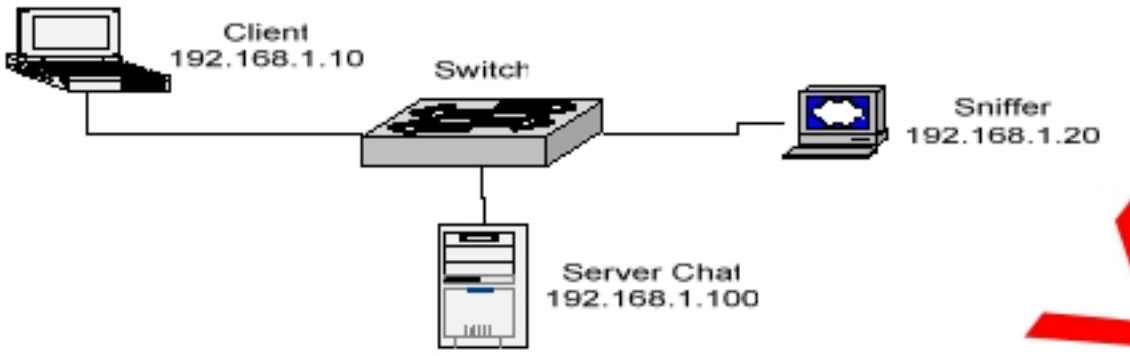
Aplikasi ini berfungsi untuk melakukan koneksi ke server chatting. Aplikasi ini hanya digunakan untuk user yang melakukan koneksi melalui web browser. Untuk mengaktifkan dan menjalankan aplikasi dilakukan melalui alamat website server chat.



Gambar 4.5 Aplikasi Web Client

Sistem pengujian yang dilakukan pada aplikasi yang dibuat dilakukan dengan menggunakan software Packet Sniffer yang dapat digunakan untuk melakukan sniffing (penyadapan) terhadap data dikirim melalui jaringan.

STIKOMP SURABAYA



Gambar 4.6 Konfigurasi pengujian

4.1.5 Pada Sniffer

Pada sisi *Sniffer* dilakukan beberapa pengujian terhadap kinerja dari aplikasi yang dibuat dengan melakukan beberapa percobaan yaitu :

A. Port Scanning

Sniffer mencoba melakukan scanning port pada jaringan menggunakan software Ether Detect Packet Sniffer untuk mencari beberapa informasi penting pada server, pada port yang sedang digunakan untuk komunikasi dengan PC *client*. Alamat ip *address* dan port yang didapatkan nantinya akan digunakan sebagai informasi untuk melakukan penyadapan data yang melalui jaringan. Dari hasil *scanning* yang telah dilakukan tersebut diketahui bahwa alamat ip yang sedang aktif di dalam jaringan dan sebagian port yang digunakan untuk mengirim data

B. Data Sniffing

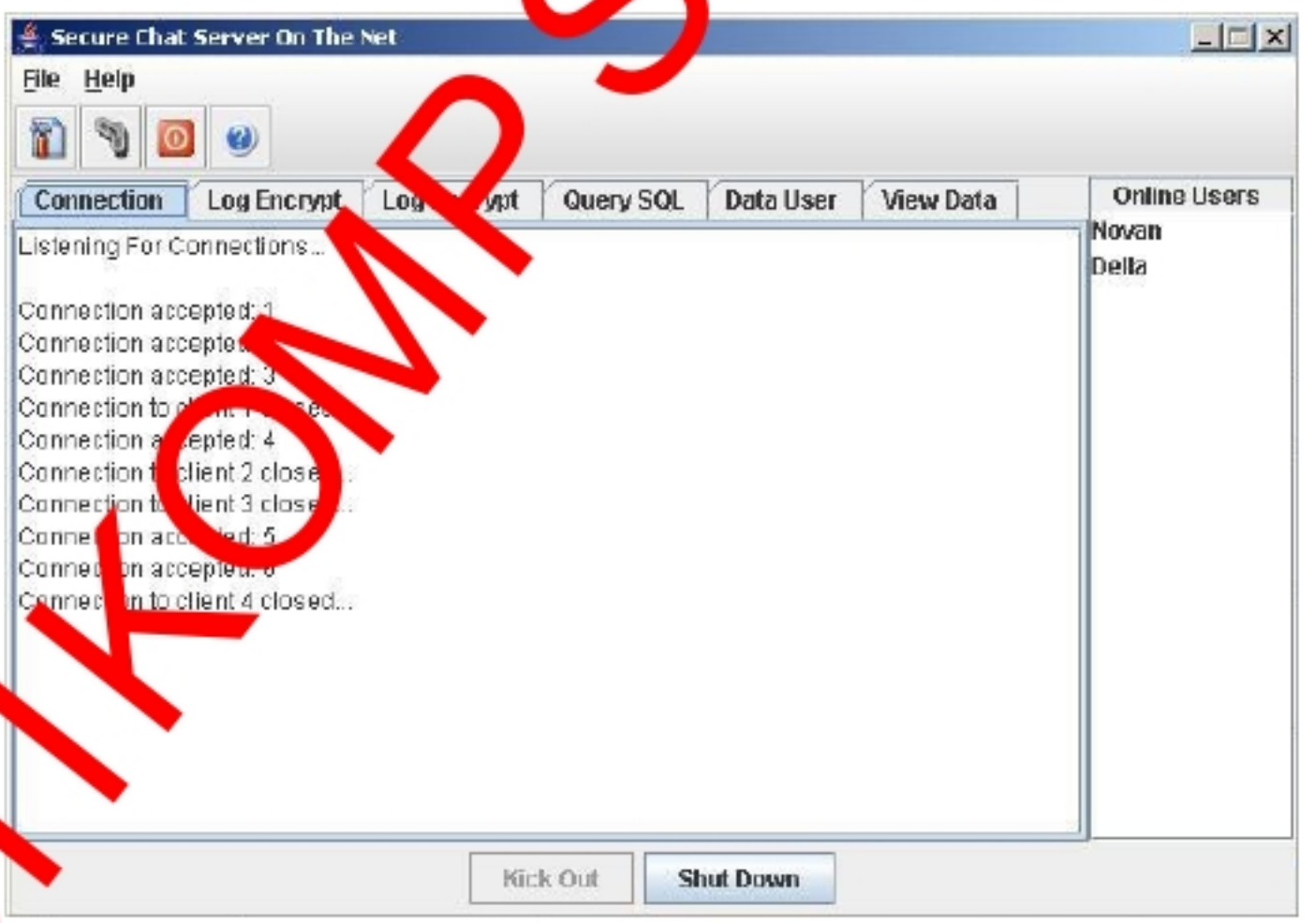
Sniffer melakukan penyadapan setiap data yang melalui jaringan dengan tujuan mendapatkan data pesan yang dikirimkan client melalui aplikasi *chatting*. Proses penyadapan dilakukan dengan menggunakan software Ether Detect Packet Sniffer.

STIKOM SURABAYA

4.1.6 Pada Aplikasi Server Chat

Semua aplikasi server yang telah berjalan akan melakukan proses monitoring secara terus – menerus dengan cara memeriksa adanaya request koneksi dari klien maupun membaca pesan yang dikirimkan klien yang telah login sebelumnya, jika sistem menemukan adanya request koneksi maka server akan meresponse dengan membuka jalur koneksi socket dan memfilter request sesuai dengan kriteria yang telah ditetapkan sebelum. Adapaun kriteria yang ditetapkan adalah :

1. Apakah jumlah klien yang terkoneksi belum memenuhi batas maksimum klien yang dapat di handle oleh server.
2. Nama login yang digunakan user apakah sudah digunakan oleh user yang sedang online.



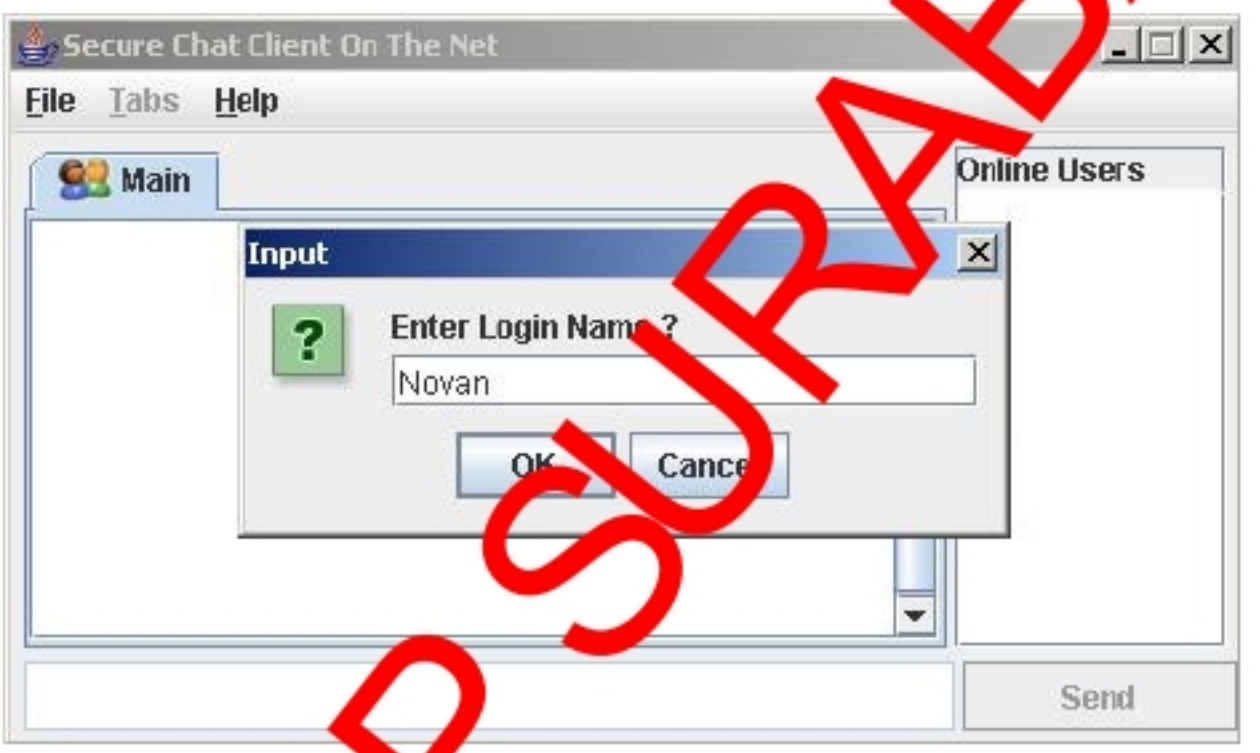
Gambar 4.7 Chat server tab connection

STIKOMP SURABAYA

4.1.7 Pada Aplikasi Chat Client (aplikasi desktop)

Client dapat berkomunikasi dengan user lain dengan apliaksi yang dibuat dengan melakukan koneksi terlebih dahulu ke server chat, caranya adalah sebagai berikut :

1. Jalankan aplikasi client, buka menu File | Connect, maka akan muncul tampilan seperti di bawah.

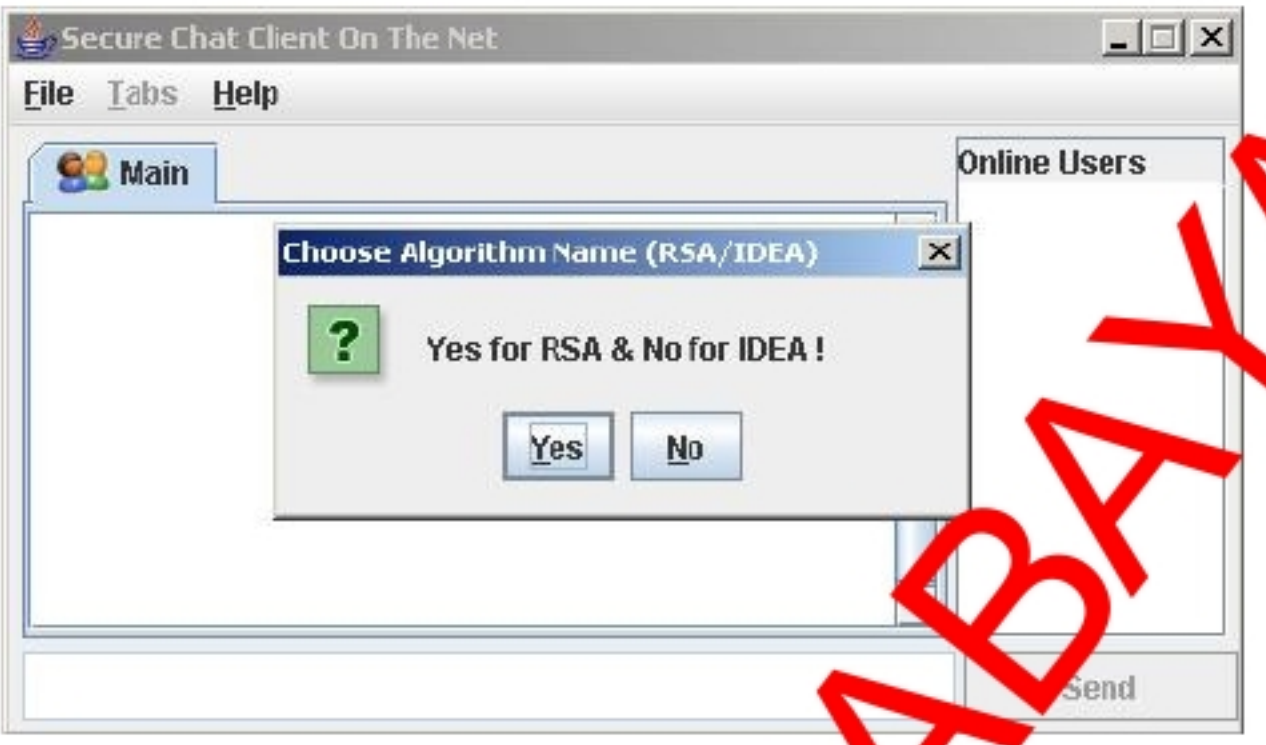


Gambar 4.8 Client input nama login

User diminta untuk memasukkan nama login yang akan digunakan sebagai pengenal ketika melakukan chatting. Setiap user harus memiliki nama login yang berbeda jika nama login yang dimasukkan sama dengan user lain yang telah ada sebelumnya, maka akan muncul pesan konfirmasi error dari aplikasi

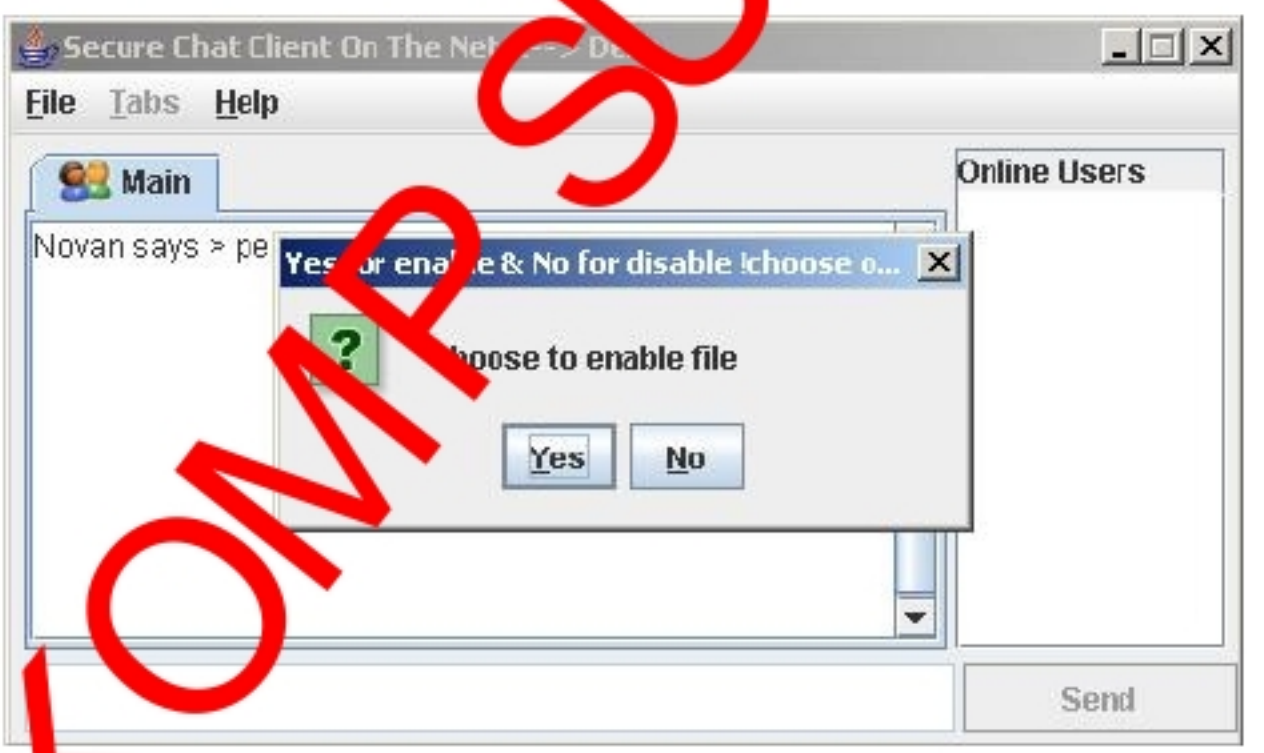
2. Pilih algoritma enkripsi yang akan digunakan.

STIKOMPS SURABAYA



Gambar 4.9 Client memilih algoritma enkripsi

3. Pilih apakah fitur pengiriman file diaktifkan atau tidak.



Gambar 4.10 Client mengaktifkan fitur file

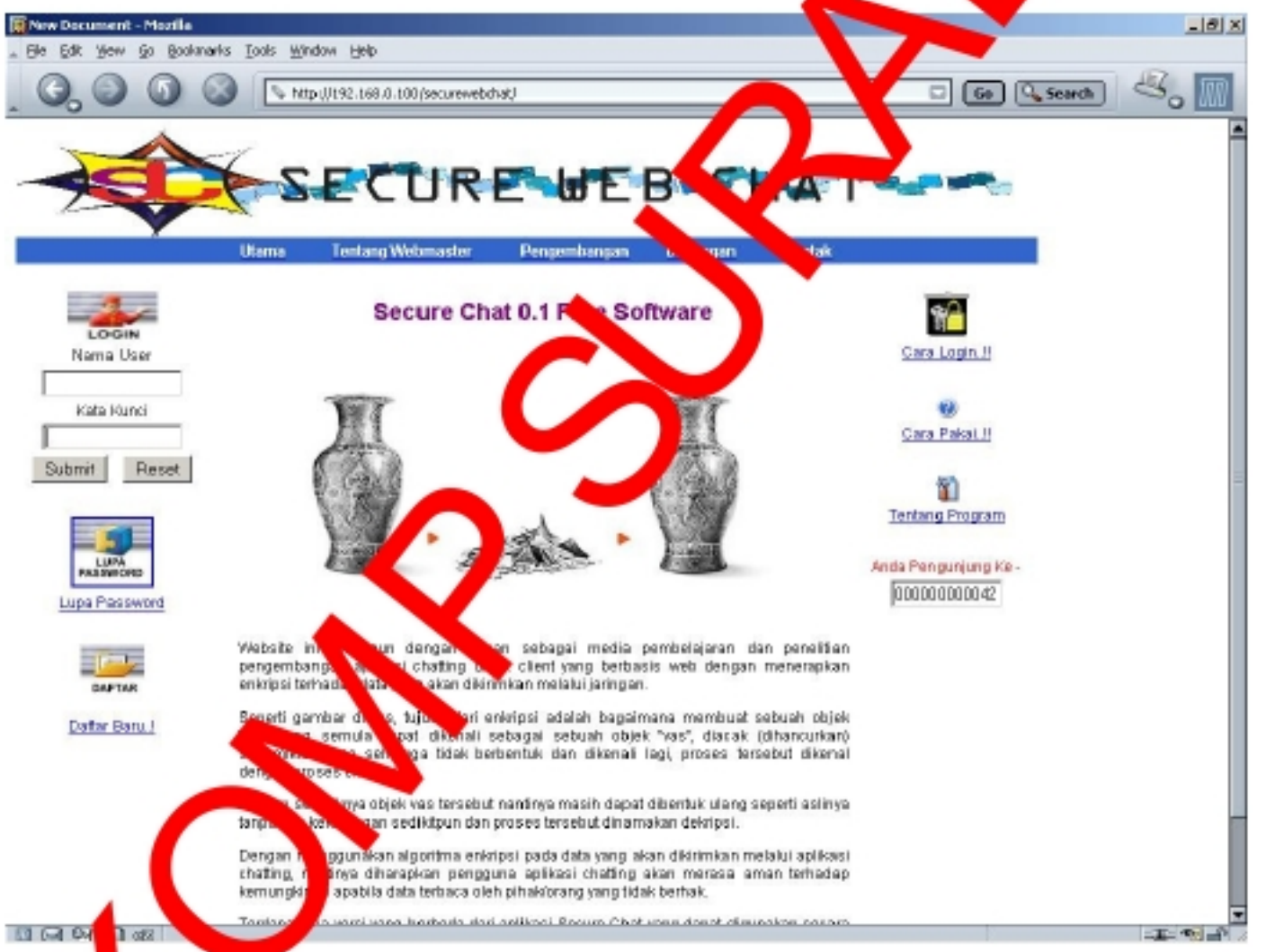
Jika koneksi berhasil maka user dapat memulai melakukan chatting dengan user lain.

STIKOM SURABAYA

4.1.8 Pada Aplikasi Chat Client (web based)

Client yang mengakses melalui web browser dapat berkomunikasi dengan user lain dengan aplikasi yang dibuat dengan melakukan koneksi terlebih dahulu ke server chat, caranya adalah sebagai berikut :

1. Jalankan aplikasi web browser, buka alamat URL http://192.168.0.100/securewebchat/, maka akan muncul tampilan seperti di bawah.



Gambar 4.11 Halaman utama secure web chat

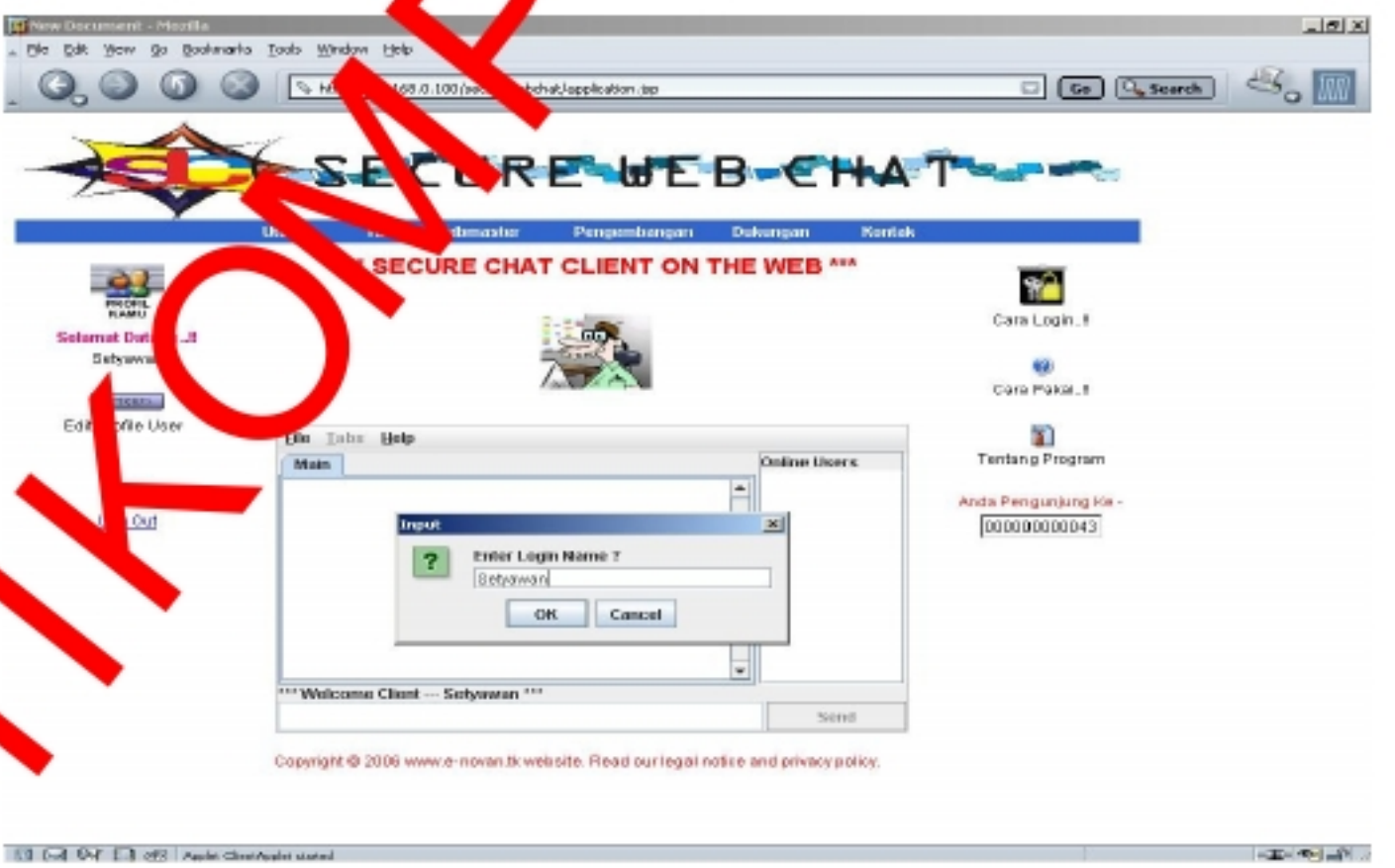
User diminta untuk memasukkan nama user dan password, apabila user dan password ditemukan maka user dapat melakukan koneksi dengan server, jika tidak ditemukan maka user akan mendapat pesan konfirmasi error.

STIKOMP.SURABAYA

2. Pada halaman user profile, pilih link aplikasi secure chat.

Gambar 4.12 Halaman user profile

3. Buka menu File | Connect, maka akan muncul tampilan seperti di bawah.

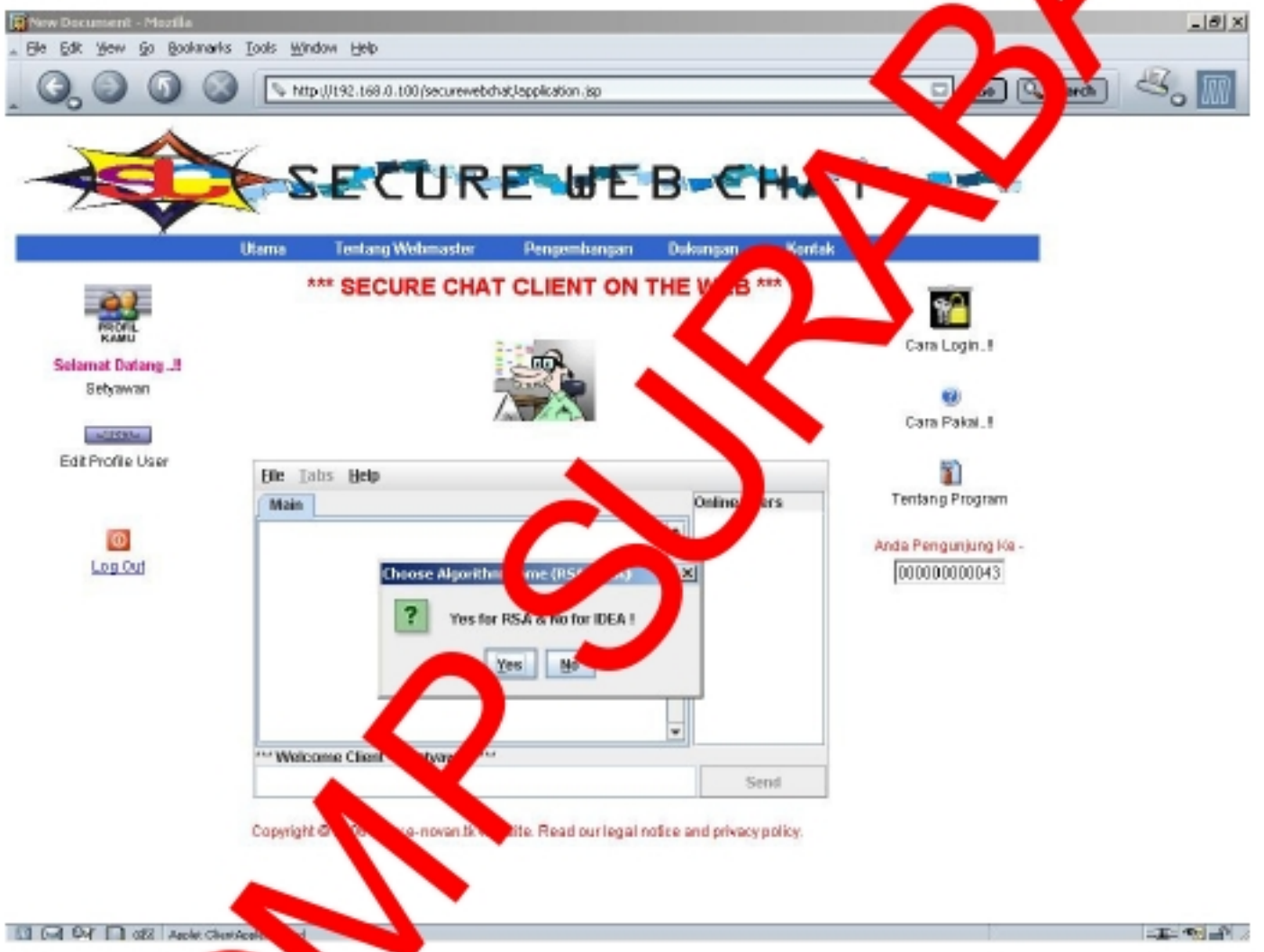


Gambar 4.13 Client web input nama login

STIKOM SURABAYA

User diminta untuk memasukkan nama login yang akan digunakan sebagai pengenal ketika melakukan chatting. Setiap user harus memiliki nama login yang berbeda jika nama login yang dimasukkan sama dengan user lain yang telah ada sebelumnya, maka akan muncul pesan konfirmasi error dari aplikasi

4. Pilih algoritma enkripsi yang akan digunakan.



Gambar 4.14 Client web memilih algoritma enkripsi

Jika koneksi berhasil maka user dapat memulai melakukan *chatting* dengan user lain.

4.2 Hasil Pengujian

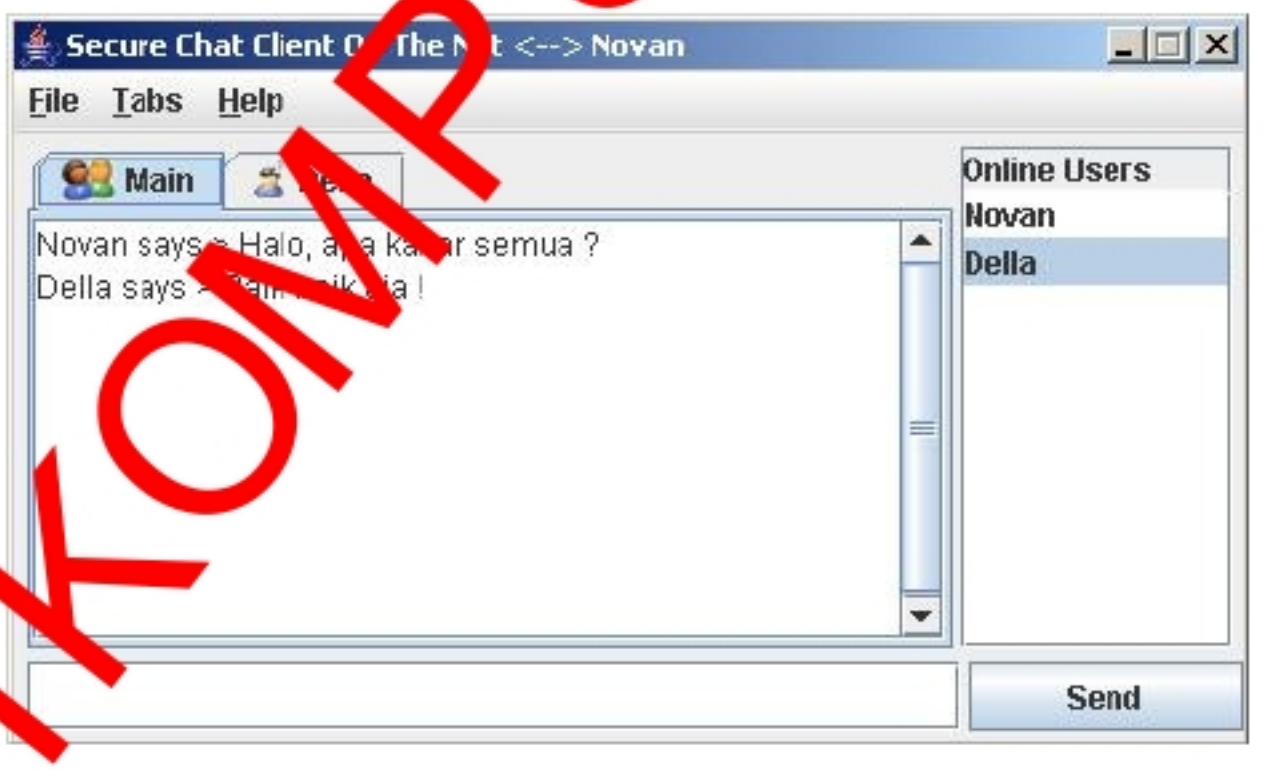
Pada proses pengujian yang dilakukan dibedakan menjadi dua kategori yaitu *sniffing* (penyadapan) data dan *performance* enkripsi. Berdasarkan proses

STIKOM SURABAYA

pengujian pada aplikasi dapat diketahui bahwa *software Ether Detect* mampu membaca data yang melewati jaringan. Proses penyadapan dilakukan pada 2 (dua) jenis aplikasi yang berbeda yaitu : aplikasi yang tidak menerapkan enkripsi data dan aplikasi yang menerapkan enkripsi data.

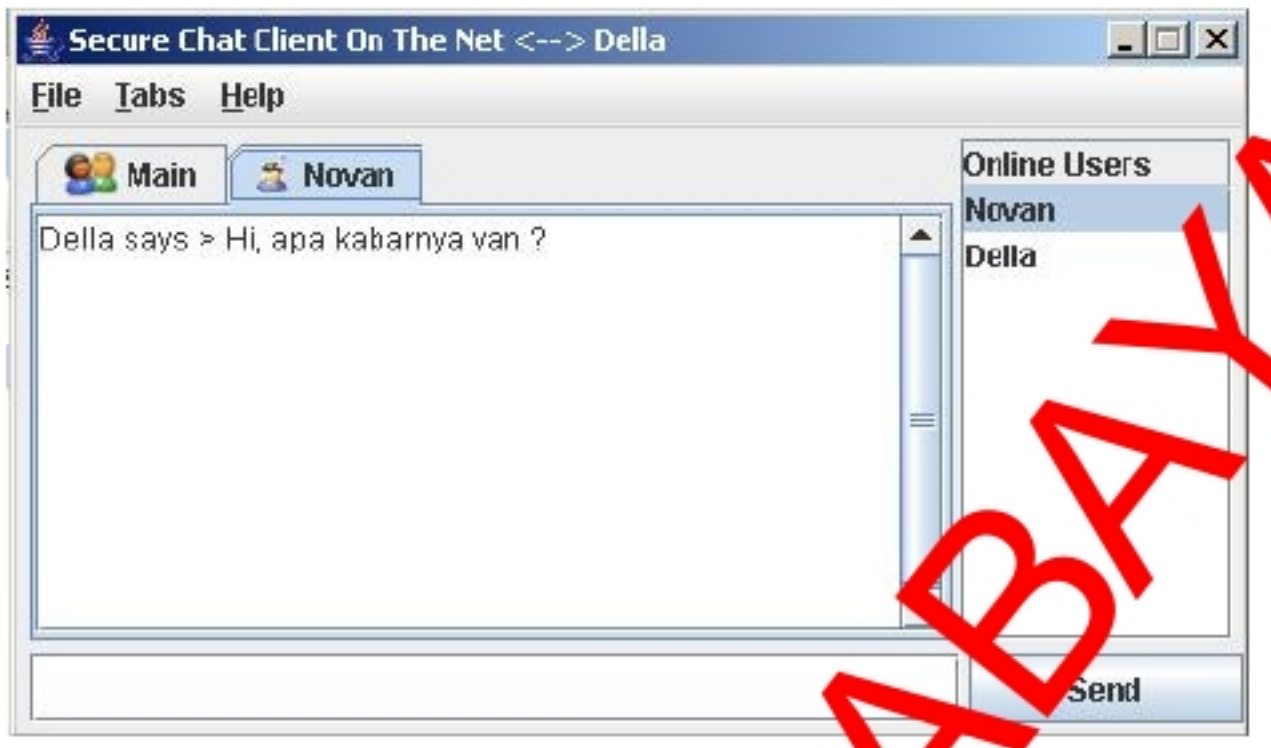
Dari penyadapan yang dilakukan akan didapatkan hasil yang berbeda, untuk aplikasi yang tidak menerapkan enkripsi data, pesan *plaintext* yang dikirim user melalui jaringan dapat dengan mudah dibaca. Adapun sebaliknya data yang dikirim user melalui jaringan dengan menggunakan aplikasi yang menerapkan enkripsi data mengakibatkan data hasil penyadapan tidak dapat dibaca.

Untuk dapat membaca data hasil penyadapan seorang sniffer harus mengetahui algoritma enkripsi dan kunci yang digunakan, tanpa mengatahui informasi algoritma dan kunci yang digunakan oleh user pengirim dan user penerima sangat sulit atau mustahil untuk mendekripsi data yang telah disadap.

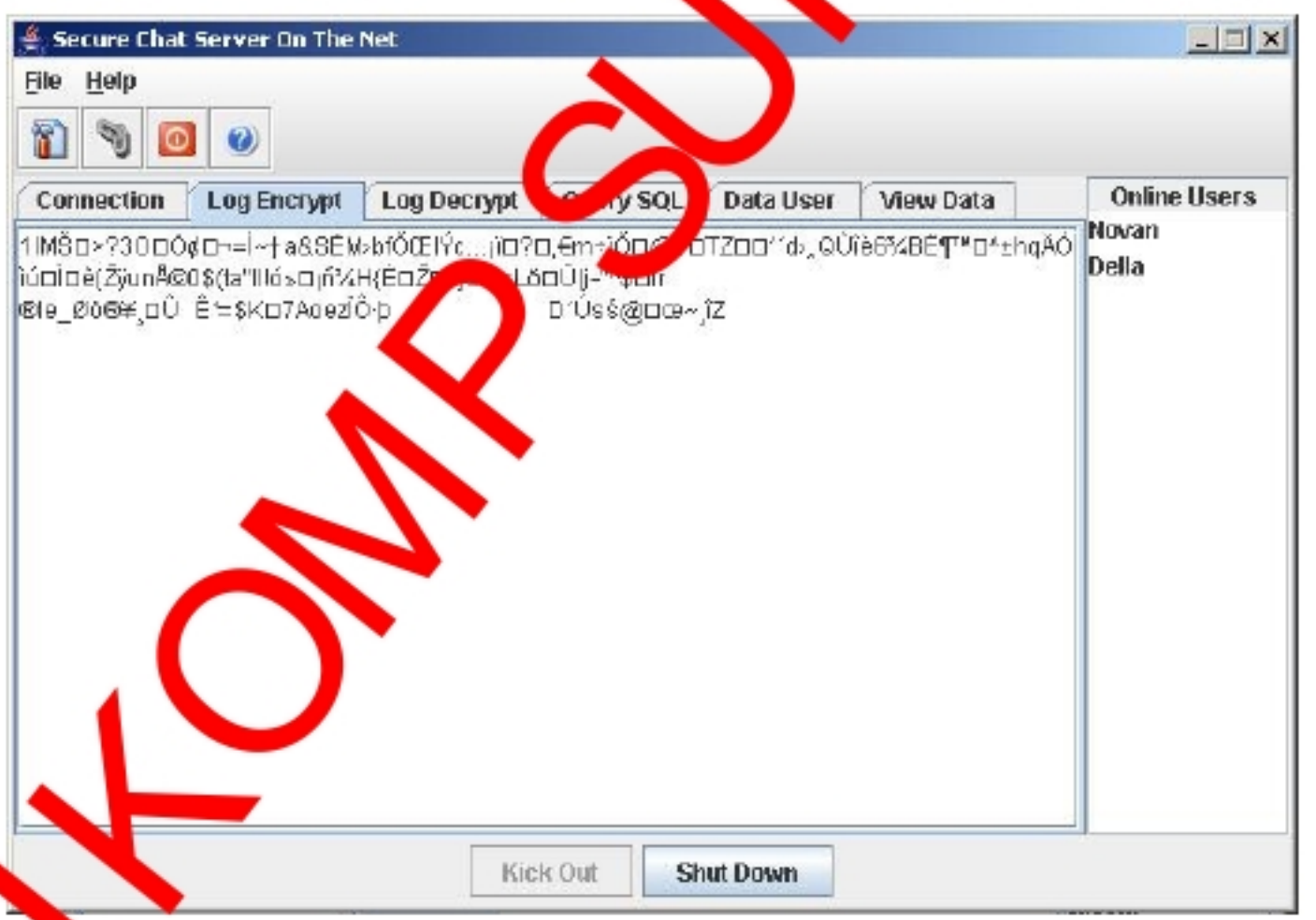


Gambar 4.15 Client kirim pesan ke seluruh user

SURABAYA
STIKOMPS



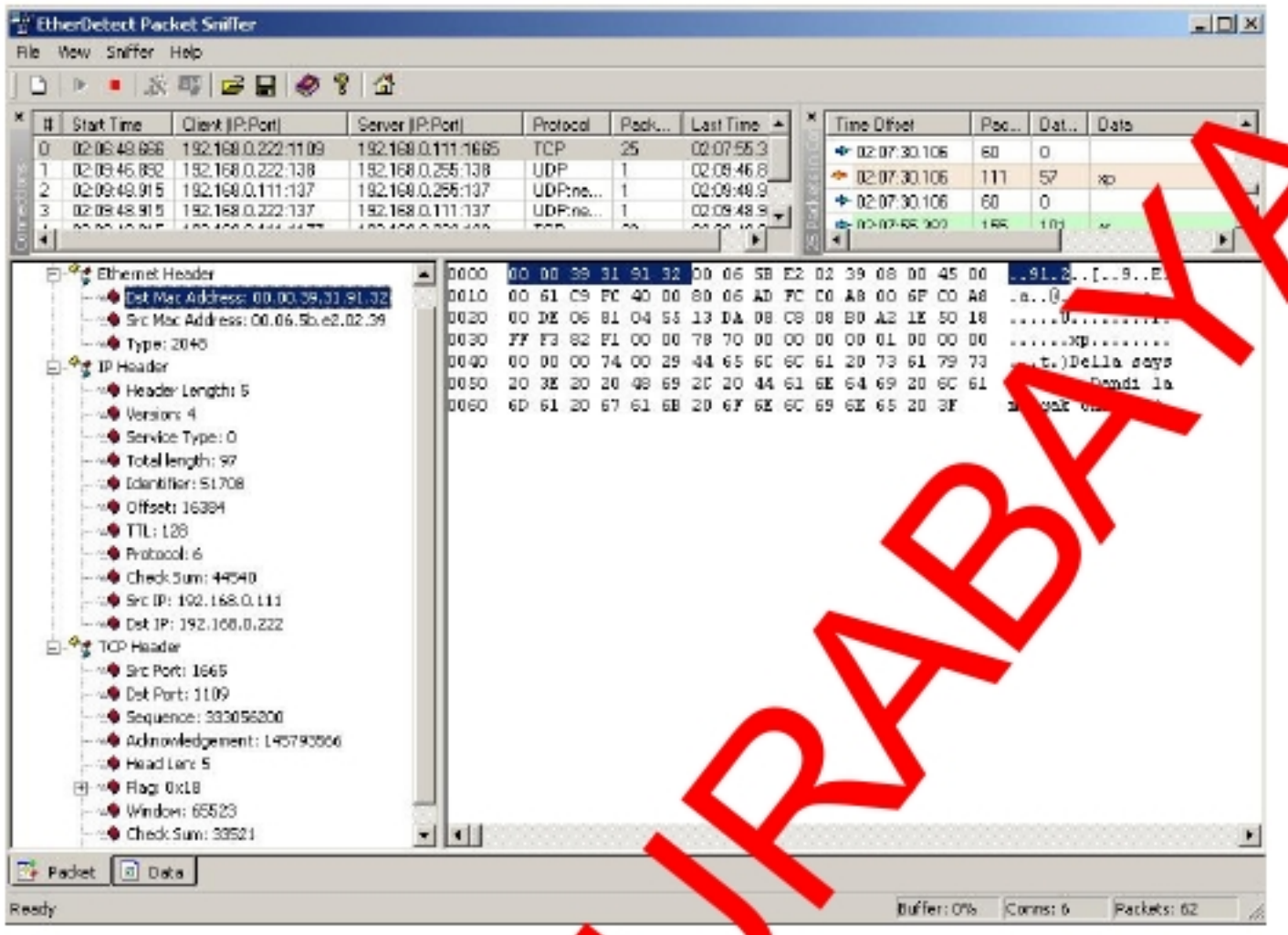
Gambar 4.16 Antarmuka untuk pengiriman pesan ke user tertentu



Gambar 4.17 Log pesan user yang telah dienkripsi

Dari gambar 4.17 dapat diketahui bahwasannya data pesan yang dikirim client telah terenkripsi.

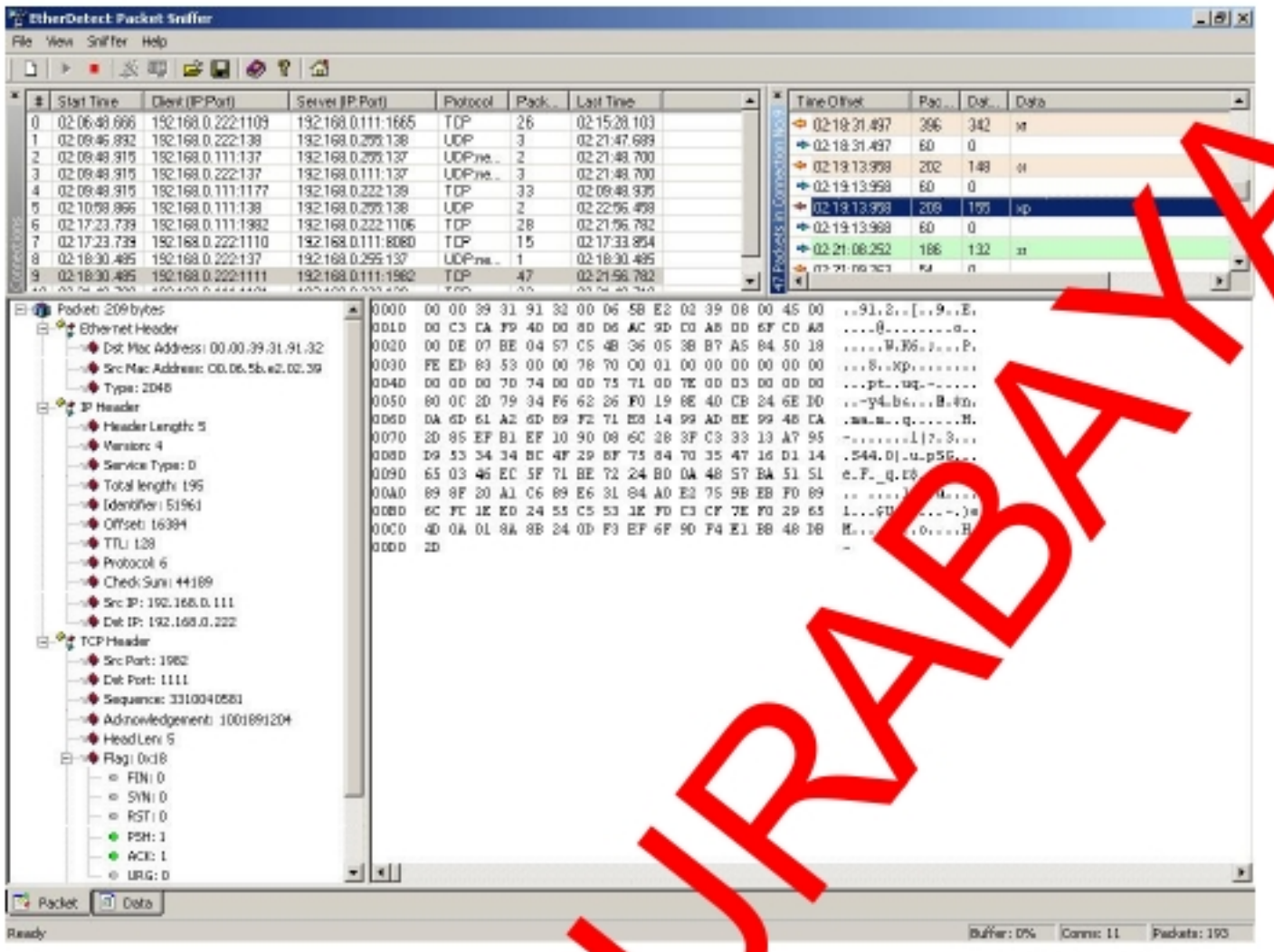
SURABAYA
STIKOMP



Gambar 4.18 Sniffing data pesan plaintext menggunakan program EtherDetect

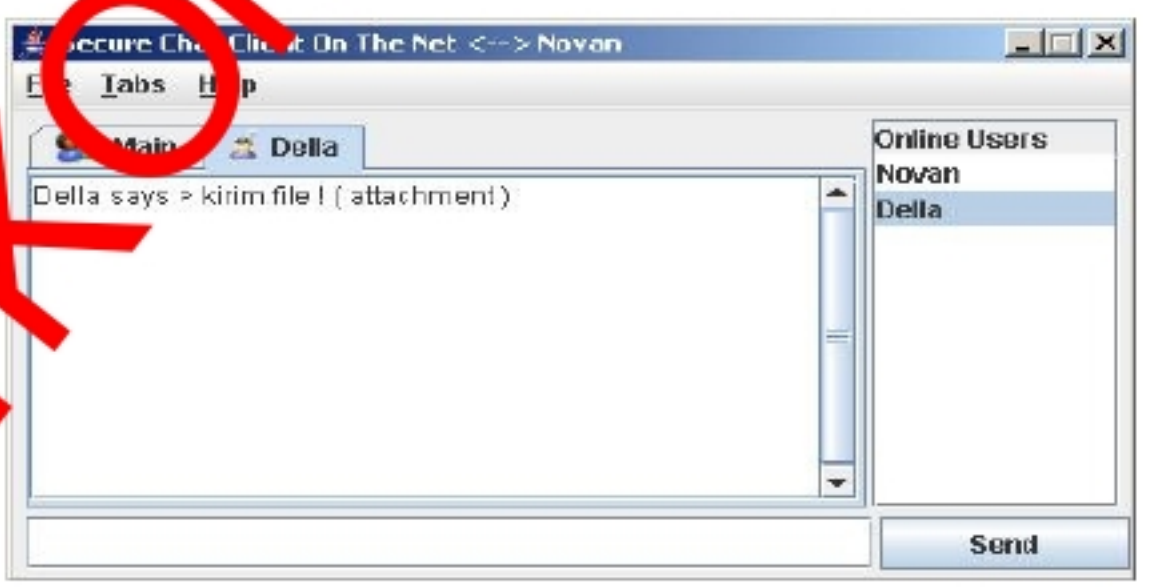
Pada gambar 4.18 menunjukkan pesan user yang dikirim melewati jaringan tanpa dienkripsi data dengan mudah disadap dan diketahui isi dari pesan user dengan mudah.

SURABAYA
STIKOMP



Gambar 4.19 Sniffing data pesan enkripsi menggunakan program EtherDetect

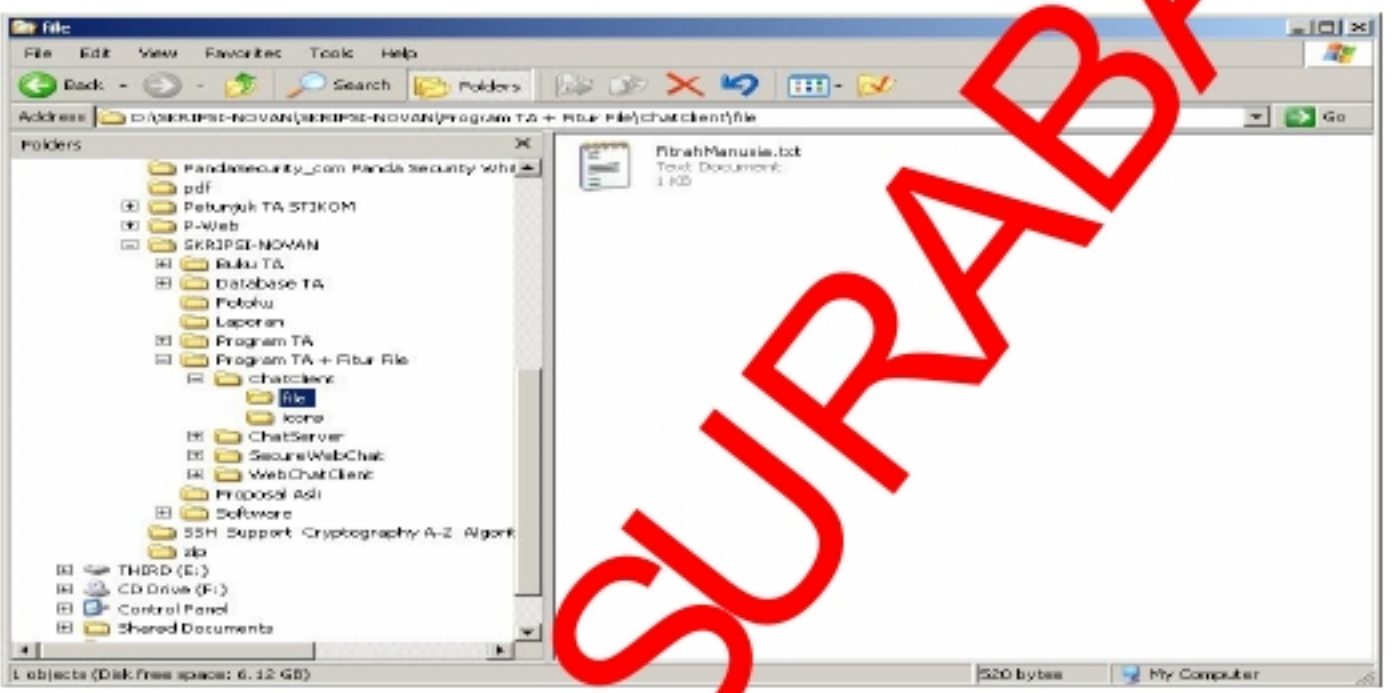
Pada gambar 4.19 proses penyadapan data pada pesan yang telah dienkripsi, tidak dapat memberikan informasi kepada sniffer, isi dari pesan yang dikirim seorang user yang melewati jaringan sulit dimengerti karena telah teracak.



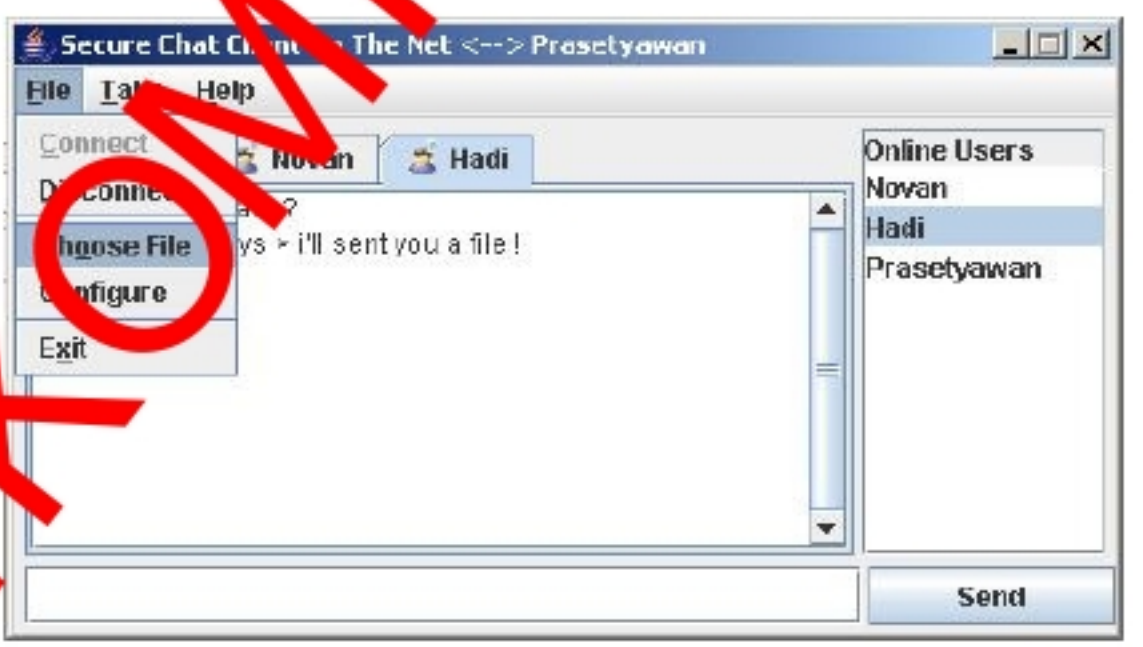
Gambar 4.20 Client kirim pesan dan file

STIKOM PGRI SURABAYA

Aplikasi yang dibuat selain dapat melakukan pengiriman pesan plaintext juga memiliki fitur untuk melakukan pengiriman file, user penerima akan menerima pesan berupa " (attachment) " yang menandakan bahwasannya data yang diterima adalah file. File yang diterima akan disimpan pada folder file yang terletak pada folder dimana aplikasi chat client berada.

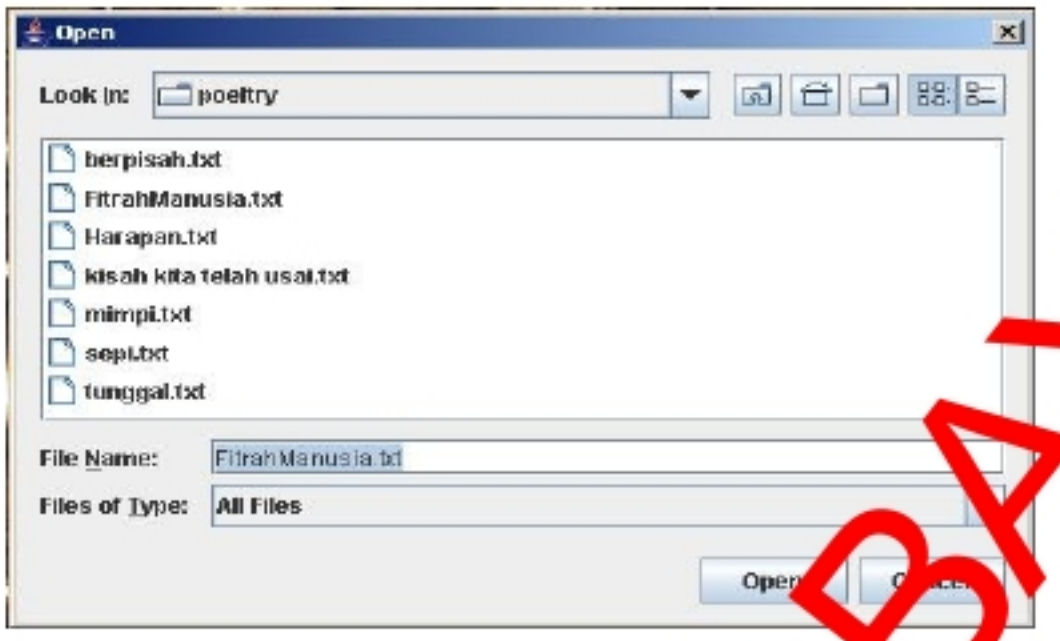


Gambar 4.21 Letak penyimpanan file yang diterima



Gambar 4.22 Client menu choose file

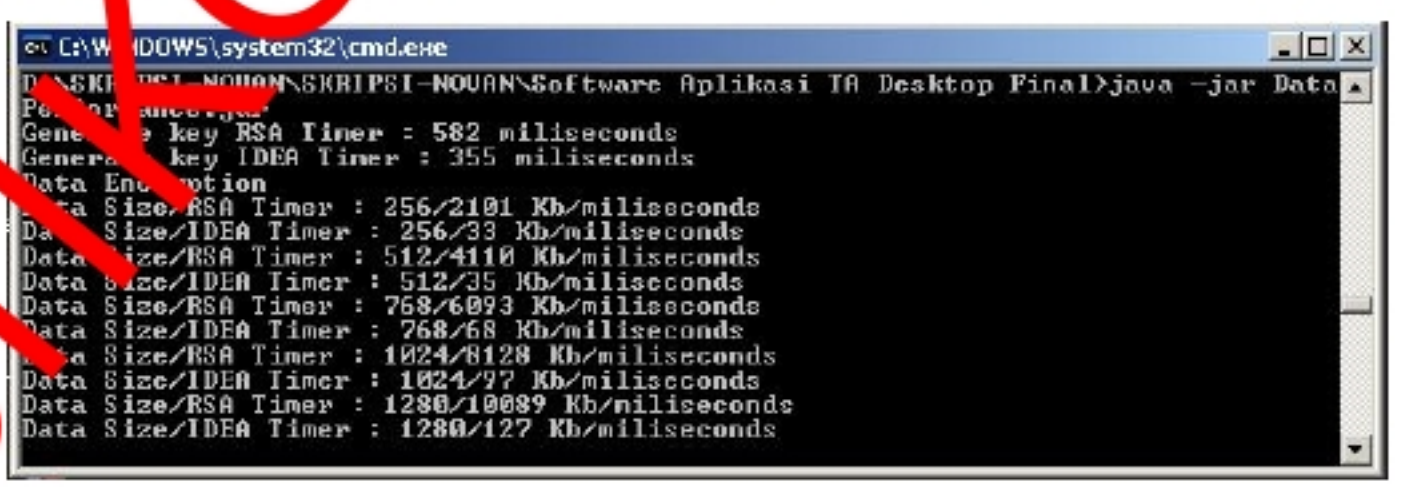
STIKOM SURABAYA



Gambar 4.23 Client menentukan file yang akan dikirim

Untuk dapat melakukan pengiriman *file* pihak user pengirim dan user penerima sebelumnya harus mengaktifkan fitur pengiriman *file*. Dari gambar diatas menunjukkan bahwa user pengirim menentukan file yang dikirim kepada user yang dituju.

Proses pengujian *performance* dari algoritma enkripsi dilakukan dengan cara menghitung waktu yang diperlukan tiap algoritma enkripsi untuk membuat kunci enkripsi dengan panjang kunci yang telah ditentukan dan kecepatan enkripsi sejumlah data dengan berbagai ukuran. Pengujian *performance* algoritma enkripsi dilakukan untuk jenis data *plaintext* dan data berupa *file.*



Gambar 4.24 Waktu proses enkripsi untuk data plaintext

STIKOM SURABAYA

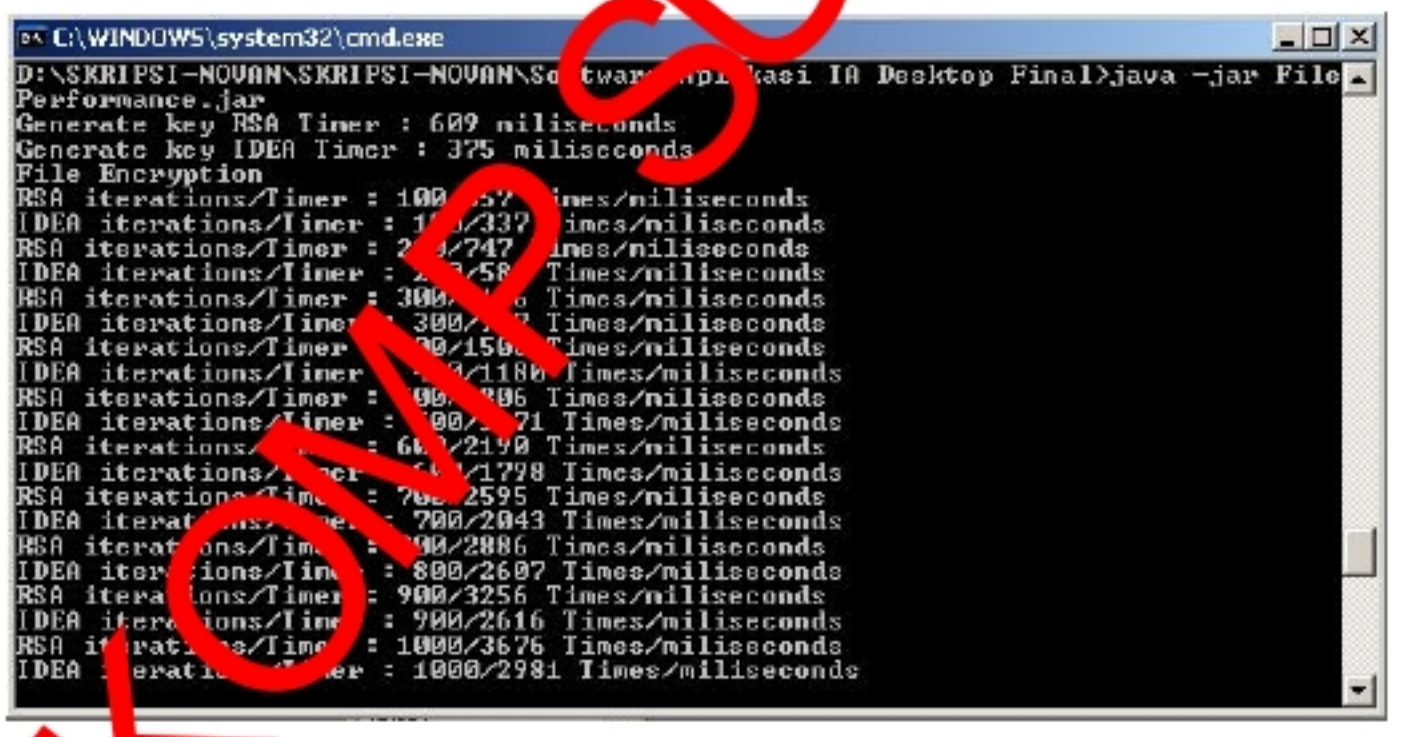
Dari gambar 4.24 dapat diperoleh waktu proses tiap-tiap algoritma enkripsi, dibawah ini adalah tabel waktu proses untuk pembuatan key dan enkripsi pada data plaintext.

Tabel 4. Waktu dibutuhkan untuk membuat key

Algorithm & Key Length (bit)	Time (millisecond)
RSA 1024 bit	582
IDEA 128 bit	355

Tabel 5. Waktu dibutuhkan untuk enkripsi data

Algorithm & Key Length (bit)	Data Size (Kb)				
	256	512	768	1024	1280
RSA 1024 bit	2101 ms	4110 ms	[illegible]	8128 ms	10089 ms
IDEA 128 bit	33 ms	35 ms	68 ms	97 ms	127 ms



Gambar 4.25 Waktu proses enkripsi untuk data file

Dari gambar 4.25 dapat diperoleh waktu proses tiap-tiap algoritma enkripsi, dibawah ini adalah tabel waktu proses untuk enkripsi pada data file.

STIKOM SURABAYA

Tabel 6. Waktu dibutuhkan untuk membuat key

Algorithm & Key Length (bit)	Time (millisecond)
RSA 1024 bit	609
IDEA 128 bit	375

Tabel 7. Waktu dibutuhkan enkripsi file

Iterasi	Algoritma Enkripsi	
	RSA	IDEA
100	357 ms	337 ms
200	747 ms	582 ms
300	1126 ms	967 ms
400	1506 ms	1180 m
500	1806 ms	1671 ms
600	2190 ms	17 ms
700	2595 ms	2043 s
800	2886 ms	260 ms
900	3256 m	6 ms
1000	3676 ms	2981 s

Proses pengujian dilaku pada komputer dengan spesifikasi PC, clock processor 1 Ghz, memory b dan sistem operasi Windows XP.

4.3 Analisa

Dari pe ng dilakukan didapatkan beberapa hasil analisa, yaitu :

1. Algori ma enkripsi RSA memiliki waktu proses yang lebih lambat jika di andingkan algoritma enkripsi IDEA untuk pembuatan key enkripsi.
2. Al enkripsi RSA memiliki waktu proses yang lebih lambat jika dibandingkan algoritma enkripsi IDEA untuk proses enkripsi data plaintext maupun file.
3. Data hasil enkripsi algoritma RSA lebih panjang jika dibandingkan algoritma IDEA (gambar 4.17).

STIKOMPS SURABAYA