BAB III

ANALISA DAN DESAIN SISTEM

Pada bab ini akan dibahas mengenai Aplikasi MenginjeksiARP Pesan Data *Client* Metode *Man In The Middle Attack* yang meliputi analisa sistem dan desain sistem.

III.1. Analisis Masalah

Adapun analisa masalah pada Aplikasi Menginjeksi ARP Pesan Data Client Metode *Man In The Middle Attack* yaitu :

1. Saat sekarang ini banyak orang selalu menggunakan aplikasi dalam pengiriman data pesan dan tidak memikirkan kata-kata apa yang baru di tulisnya, dan tidak ada mengetahui isi dari percakapan tersebut.
2. Dengan menggunakan *firewall* pada jaringan LAN sulit masuk di dalam jaringan tersebut, oleh karena itu perlu melakukan injeksi ARP terhadap pengalamatan *networking*.

   Berdasarkan analisa diatas maka penulis telah melakukan evaluasi dari sistem yang sedang berjalan dan penulis menemukan kelemahan sistem yang ada. Dengan demikian penulis memberikan suatu solusi yang diharapkan dapat mengatasi kelemahan sistem yang ada. Adapun solusi yang ditawarkan adalah membangun Aplikasi Menginjeksi ARP Pesan Data *Client* Metode *Man In The Middle Attack.* Aplikasi ini adalah salah satu alat yang diyakini

mampu menangani masalah diatas yaitu dengan menggunakan *metode man in the middle attack* dapat mengontrol isi dari pembicaraan mereka

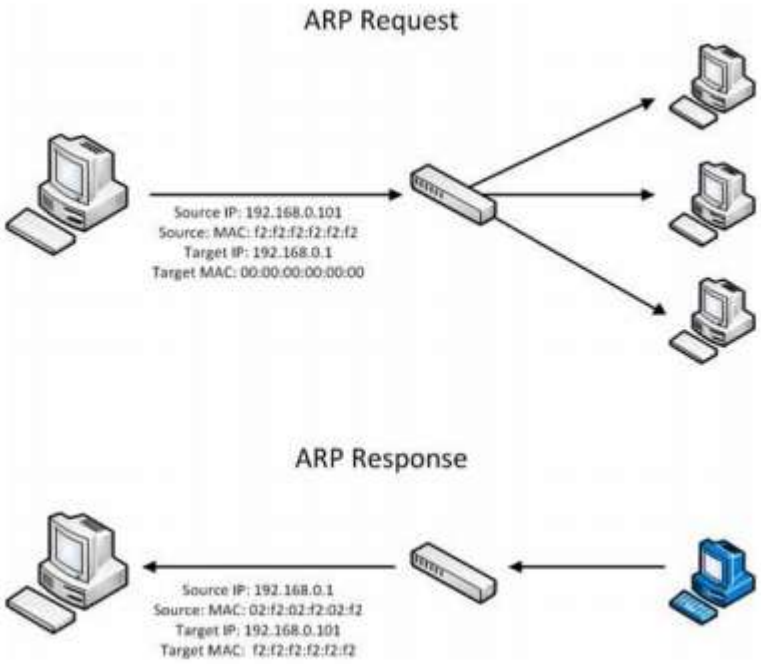
III.2. Analisa Kebutuhan *Hardware* Dan *Software*

Dalam perancangan aplikasi transfer *file* dengan *handphone*, ada beberapa perangkat yang penulis gunakan agar aplikasi berjalan sebagaimana mestinya, yaitu sebagai berikut :

1. *Hardware* (Perangkat Keras)
   a. PC (*Personal Computer*) atau laptop dengan *processor* intel *core* duo.
   b. *Memory* 2 GB.
   c. *Harddisk* 320 GB HDD.
   d. *Hub switch*
2. *Software* (Perangkat Lunak)
   a. *Windows* 7 sebagai sistem operasi.
   b. Menggunakan bahasa pemograman *Visual Basic. Net*.

III.2.1. Konsep Penyerangan metode *man-in-the-middle* dengan *ARP Poisoning*

Prinsip dari metode ini adalah untuk mengirimkan pesan ARP palsu (*spoofed*) ke sebuah Ethernet LAN. Biasanya, tujuan dari metode ini adalah untuk mengasosiasikan alamat MAC dari penyerang dengan alamat IP dari node lain (seperti *default gateway*). Setiap *traffic* yang melewati alamat IP tersebut akan disengajakan melewati penyerang dan kemudian sang penyerang bisa memilih untuk melanjutkan *traffic* ke default *gateway* yang sebenarnya atau mengubah data terlebih dahulu sebelum melanjutkan pesan tersebut. (Karunia Ramadhan, 2011)



Gambar III.1 Model ARP
Sumber : (Karunia Ramadhan; 2011)

Penyerangan metode *man-in-the-middle* dengan *ARP Poisoning* ini akan dilakukan dengan bantuan beberapa perangkat lunak :

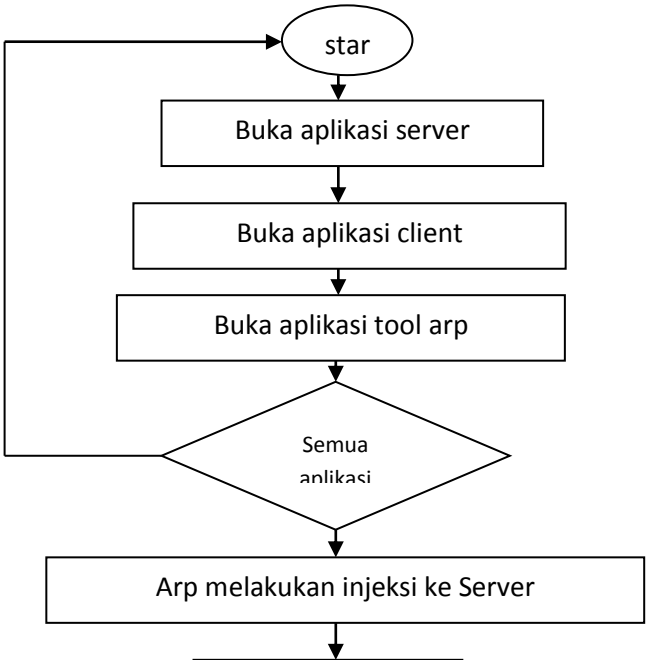
1. NMAP : digunakan untuk mendapatkan nama *host* dan alamat MAC dari IP yang ada pada jaringan.
2. CAIN : perangka lunak yang bertujuan untuk mencari kata kunci yang hilang dengan cara menyadap jaringan, *brute-force,* penyerangan secara *cryptalanisis* dan lain lain.
3. *Wireshark* : perangkat lunak untuk menganalisis paket-paket yang ada pada jaringan.

Mesin tes yang digunakan adalah sebuah desktop PC dengan sistem operasi Windows 7 sebagai penyerang dan sebuah mesin virtual dengan sistem operasi Ubuntu 9.04 sebagai korban. Aturan jaringan dari mesin virtual dibuat menjadi sistem *bridging*. Hal itu kemudian menyebabkan mesin virtual mampu menyambung dalam jaringan LAN yang sudah ada dengan IP seperti mesin lainnya.

Alamat IP penyerang : 192.168.1.110

Alamat IP korban : 192.168.1.100

Pengujian dilakukan dengan menginterupsi koneksi IP korban dengan *gateway* jaringan LAN ke internet. Dengan kata lain, penyerang hanya melakukan *poisoning* pasif terhadap korban dan melihat informasi yang dikirimkan oleh korban kepada *gateway* (192.168.1.1), meneruskannya, dan mengembalikannya kepada korban. Untuk melakukan penyerangan *man-in-the-middle-attack* yang lebih aktual dan secara langsung mengubah data, penyerang harus mengetahui kedua IP korban dan bertindak sebagai perantara diantara mereka, dan mengubah paket data yang dikirimkan sesuai tujuan. Sayangnya untuk mengubah paket data yang dikirimkan dan secara langsung akan memerlukan penelitian lebih lanjut tentang protokol jaringan dan komunikasi yang digunakan.



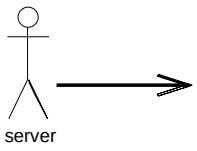
Gambar III.2 *Flowchart* Penyerangan metode *man-in-the-middle* dengan *ARP Poisoning*

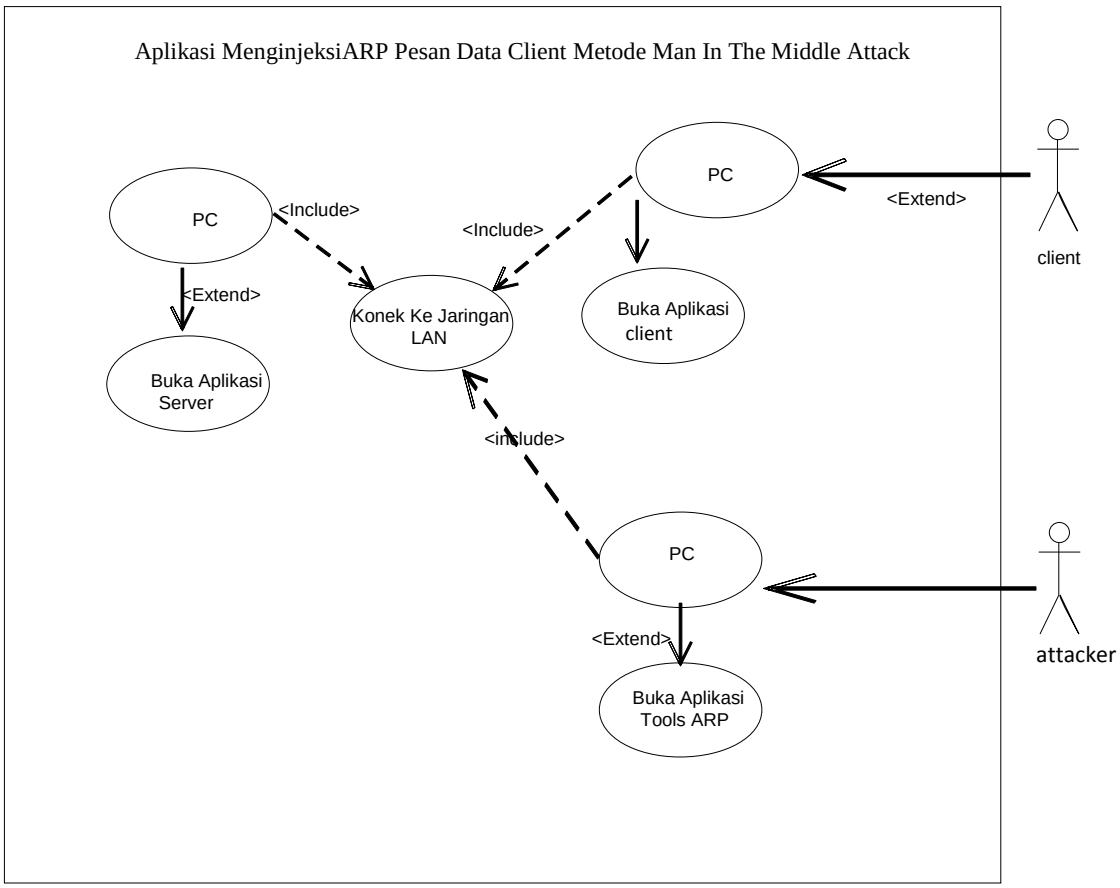
III.3. Desain Sistem

Perancangan desain sistem yang akan dibangun menggunakan pemodelan *Unified Modeling System* (UML). Diagram-diagram yang digunakan *use case* diagram, *activity* diagram, *class* diagram dan *squence* diagram.

III.3.1. *Use Case*

Use case diagram berfungsi untuk menggambarkan kegiatan aktor atau pengguna aplikasi. Adapun *use case* diagram aplikasi yang dirancang dapat dilihat pada gambar III.3. berikut.



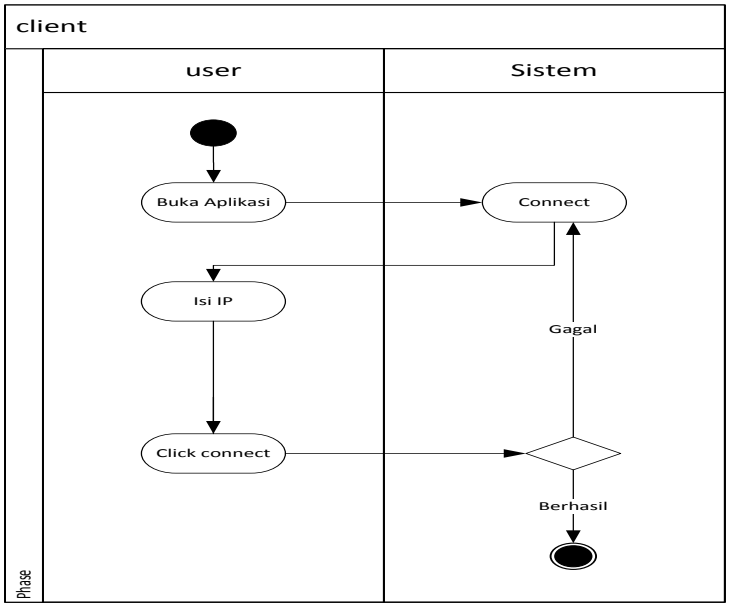


Gambar III.3 *Use Case* Diagram

III.3.2. *Activity* Diagram

1. *Activity* Diagram client

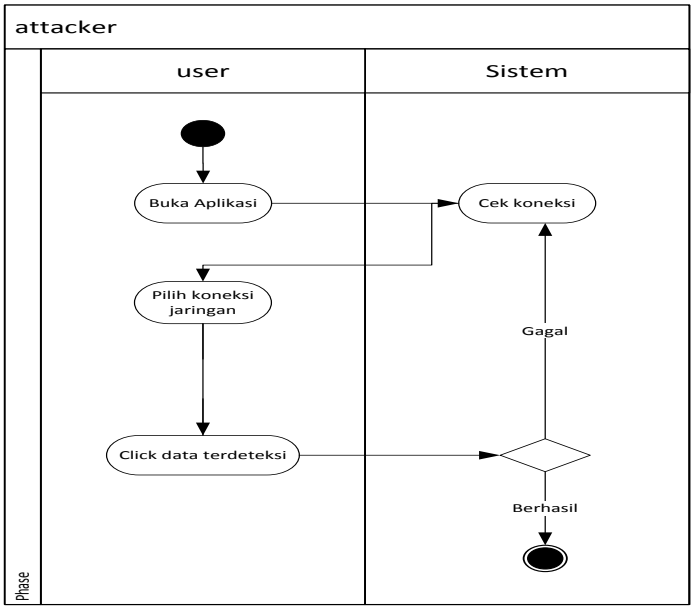
Berikut gambar *activity diagram* untuk *clien t*pada Aplikasi Menginjeksi ARP Pesan Data *Client* Metode *Man In The Middle Attack.*



Gambar III.4 ***Activity Diagram Client***

2. *Activity* Diagram Attacker

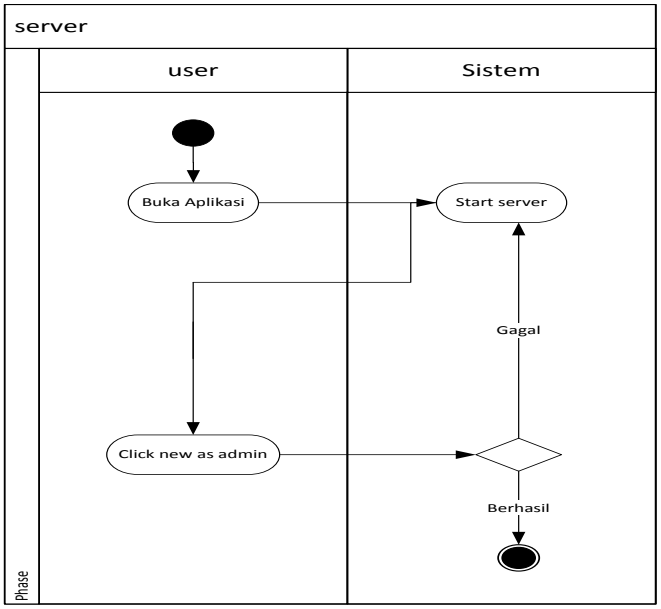
Untuk proses penyerangan (*attacker*)berikut gambar*activity diagram* untuk *attacker* pada Aplikasi Menginjeksi ARP Pesan Data *Client* Metode *Man In The Middle Attack.*



Gambar III.5 ***Activity Diagram Attacker***

3. *Activity* Diagram server

Server akan menerima segala *response* yang diberikan oleh *client* berikut gambar *activity diagram* untuk server pada Aplikasi Menginjeksi ARP Pesan Data *Client* Metode *Man In The Middle Attack.*

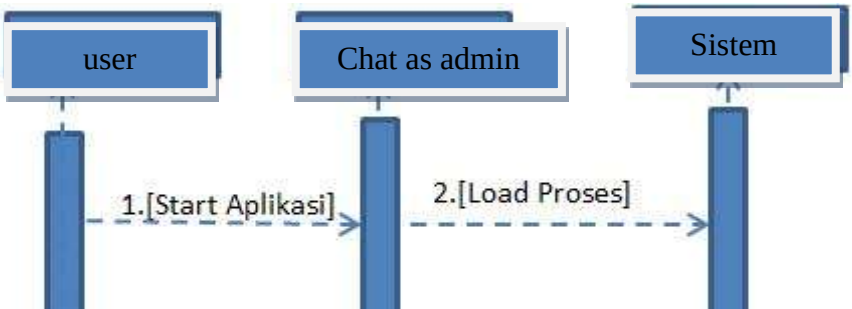


Gambar III.6 ***Activity Diagram Server***

III.3.3. *Squence Diagram*

Sequence diagram digunakan untuk menggambarkan perilaku pada sebuah skenario proses penggunaan aplikasi. Berikut ini adalah *Sequence* diagram aplikasi yang dirancang.

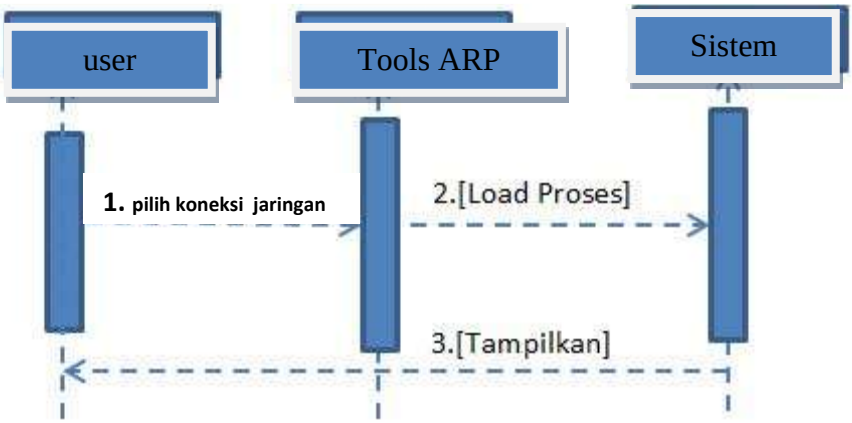
1. *Sequnce* Diagram *Server* Aplikasi, untuk diagram *Server* aplikasi dapat dilihat pada gambar III.7. dibawah ini.



1. Start server

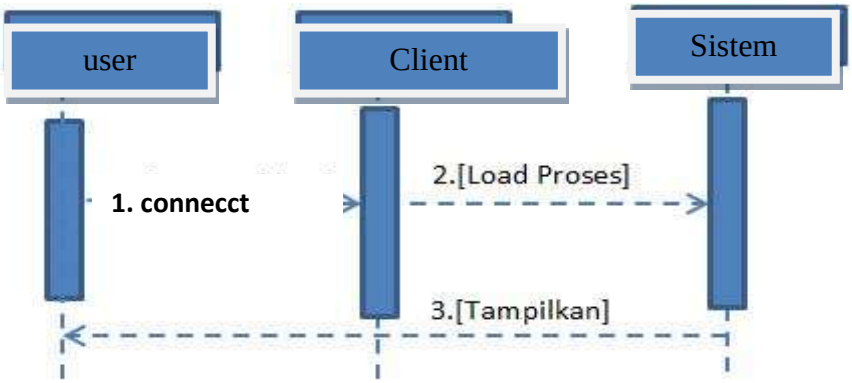
Gambar III.7 *Sequence Diagram server*

2. *Sequnce* Diagram A*ttacker* Aplikasi, untuk diagram *Server* aplikasi dapat dilihat pada gambar III.8. dibawah ini.



Gambar III.8 *Sequence Diagram Attacker*

3. *Sequnce* Diagram *Client* Aplikasi, untuk diagram client aplikasi dapat dilihat pada gambar III.9. dibawah ini.

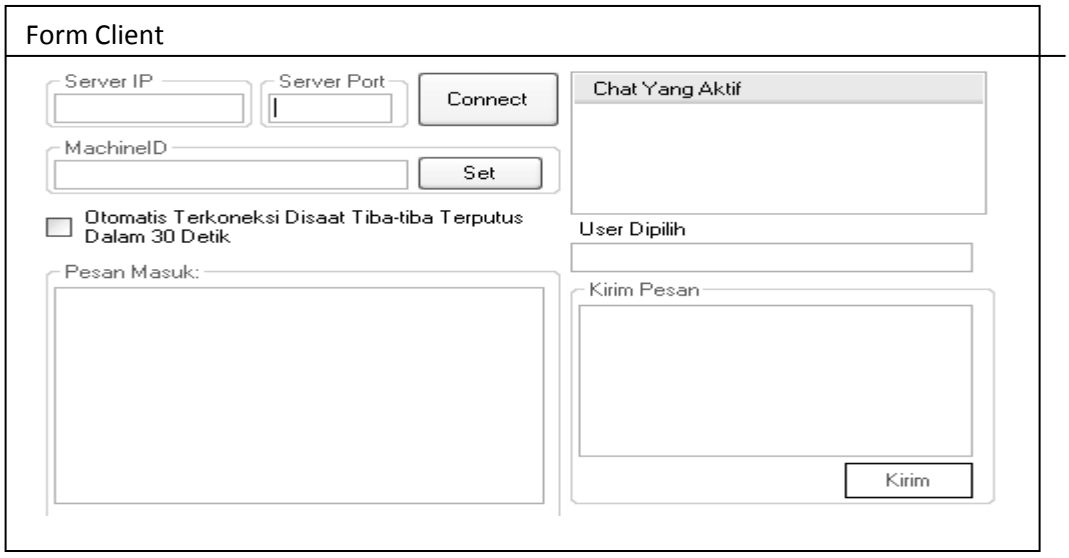


Gambar III.9 ***Sequence Diagram Client***

III.4. Disain Sistem Secara Detail

1. Tampilan Form *Client*

Tampilan sistem ini akan dijelaskan mengenai rancangan aplikasi yang akan dikerjakan serta fitur-fitur yang akan dipakai pada aplikasi tersebut.



Gambar III.10 Tampilan Form *Client*

Adapun pada perancangan tampilan form *client* dapat dilihat pada uraian berikut:

Pada kolom *Server IP* menggunakan *textbox*

Pada kolom *Server Port* menggunakan *textbox*

Pada kolom *Machine ID* menggunakan *textbox*

Pada kolom Pesan Masuk menggunakan *textbox*

Pada kolom Kirim Pesan menggunakan *textbox*

Pada kolom User Yang Dipilih menggunakan *textbox*

Pada kolom Chat Yang Aktif menggunakan *listbox*

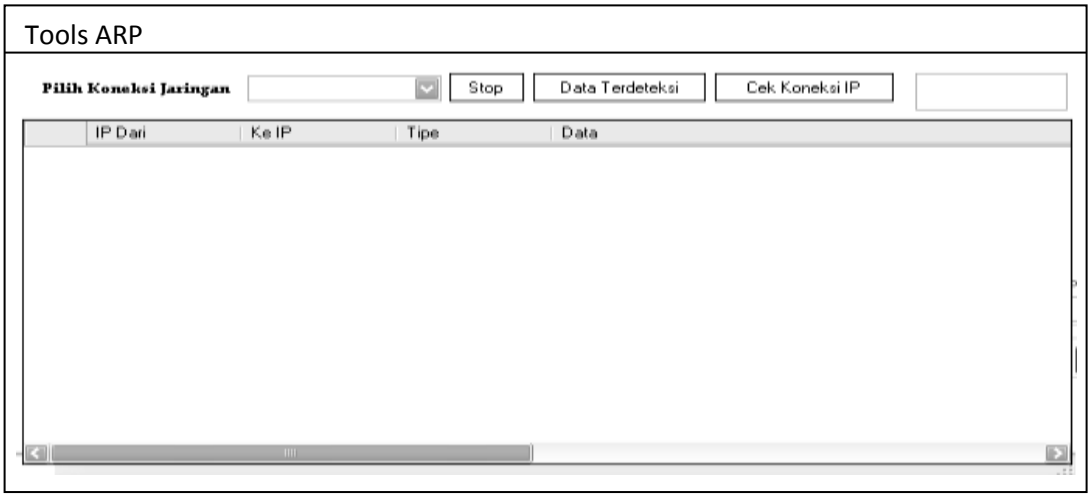
Pada tombol *connect* menggunakan *button*

Pada tombol *Set* menggunakan *button*

Pada tombol Kirim menggunakan *button*

2. Tampilan Form *Attacker*

Tampilan sistem ini akan dijelaskan mengenai rancangan aplikasi yang akan dikerjakan serta fitur-fitur yang akan dipakai pada aplikasi tersebut.



Gambar III.11 Tampilan Form *Attacker*

Adapun pada perancangan tampilan form *Attacker* dapat dilihat pada uraian berikut:

Pada label Pilih Koneksi Jaringan menggunakan *label*

Pada kolom Pilih Koneksi Jaringan menggunakan *combobox*

Pada tombol *Stop* menggunakan *button*

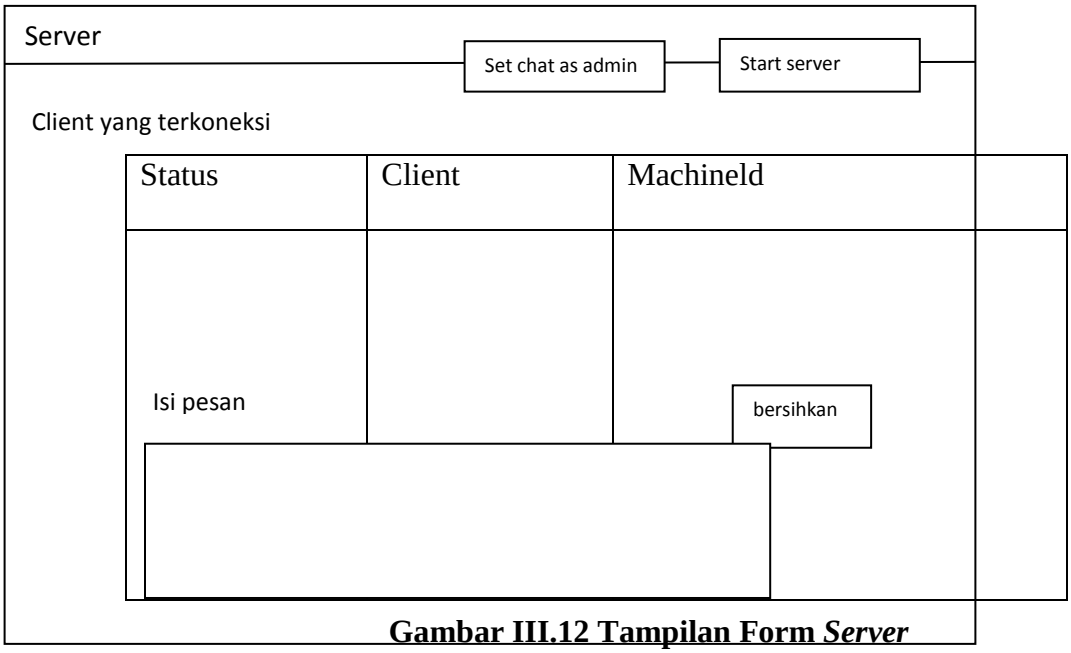
Pada tombol Data Terdeteksi menggunakan *button*

Pada tombol Cari Koneksi IP menggunakan *button*

Pada kolom IP Dari, Ke IP menggunakan *datagrid*

3. Tampilan Form Server

Tampilan sistem ini akan dijelaskan mengenai rancangan aplikasi yang akan dikerjakan serta fitur-fitur yang akan dipakai pada aplikasi tersebut.



Gambar III.12 Tampilan Form *Server*

Adapun pada perancangan tampilan form *Server* dapat dilihat pada uraian berikut:

Pada tombol *Start Server* menggunakan *button*

Pada tombol *Set Chat As Admin* menggunakan *button*

Pada kolom *Client* Yang Terkoneksi menggunakan *datagrid*

Pada kolom isi pesan menggunakan *datagrid*

Pada tombol Bersihkan menggunakan *button*