

BAB III

ANALISA DAN PERANCANGAN SISTEM

III.1. Analisis Masalah

Sniffing adalah kegiatan menyadap dan atau menginspeksi paket data menggunakan *sniffer software* atau *hardware* di *internet*. Kegiatan ini sering disebut sebagai serangan sekuriti pasif dengan cara membaca data yang berkelieran di *internet*, dan memfilter khusus untuk *host* tujuan tertentu. Jadi kegiatan ini tidak melakukan apa-apa terhadap data, tidak merubah dan tidak memanipulasi. Cukup "menyadap". Ia digunakan untuk mendapatkan informasi seperti *password*, data-data rahasia dan lainnya. Sering digunakan para *analyst networking*, baik dari kalangan *developer* maupun *network administrator*, untuk melakukan *troubleshooting*. Paket data adalah salah satu istilah yang digunakan untuk data yang keluar masuk pada sebuah perangkat komputer yang terhubung pada jaringan. Teknologi telekomunikasi yang sangat cepat akan menghadirkan beragam bentuk layanan bagi konsumen. Hal tersebut berbanding lurus dengan bisnis dan pengembangan aplikasi. Dengan peningkatan ukuran dan jumlah aplikasi perangkat jaringan maka akan semakin tinggi juga resiko gangguan yang akan terjadi terhadap jaringan. Manajemen jaringan terutama sistem monitoring menjadi sesuatu yang penting dilakukan untuk mencegah atau mengurangi gangguan yang sering terjadi. Adapun cara kerja dari *sniffing* paket data adalah :

1. Mengetahui segala bentuk informasi.
2. Mengetahui apa saja yang terjadi pada jaringan.
3. Mengetahui penyebab dari gangguan sistem jaringan (*troublehoosting*).

III.2. Kebutuhan *Hardware* dan *Software*

Adapun spesifikasi aplikasi yang dirancang memiliki kemampuan sebagai berikut:

- 1 Melakukan pemindaian *IP Address* pada *Local Area Network* dengan lingkup *workgroup* yang sama.
- 2 Dari hasil pemindaian *IP Address* dapat diketahui status dari *host* / komputer yang aktif ataupun *non* aktif.
- 3 Melakukan pemindaian terhadap *port-port* yang terbuka dari *IP Address* yang aktif.

III.3. Lingkungan Operasi

Untuk membangun aplikasi *Sniffer* Paket Data *Networking* untuk memonitoring jaringan sesuai dengan spesifikasi kebutuhan, dibutuhkan lingkungan operasi sebagai berikut,

1. Perangkat lunak "*Software*", kebutuhan perangkat lunak yang digunakan terdiri dari :
 - a. Sistem Operasi *Windows 7*.
 - b. *Java SDK*.
 - c. *NetBeans*.
2. Perangkat kerast "*Hardware*", kebutuhan perangkat keras yang digunakan terdiri dari :
 - a. Jaringan LAN.
 - b. Komputer minimal setara dengan *Pentium 4* (empat).
 - c. *Mouse.Keyboard*, dan Monitor.

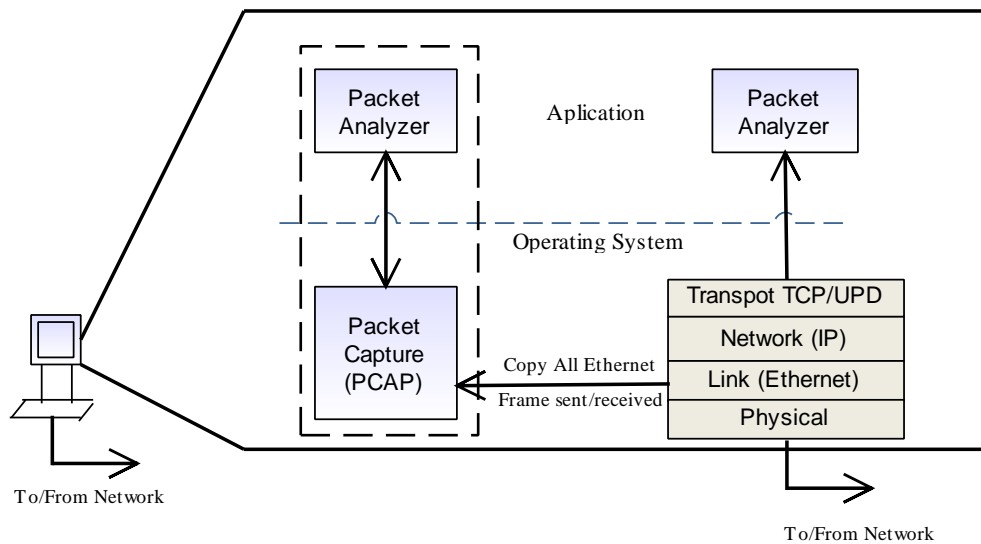
III.4. Perancangan Aplikasi

Pada perancangan aplikasi *sniffer* paket data *networking* untuk memonitoring jaringan ini akan dijelaskan mengenai rancangan aplikasi yang akan dikerjakan, dari rancangan *interface*, algoritma perancangan, dan rancangan keluaran serta fitur-fitur yang akan dipakai pada aplikasi tersebut.

III.4.1. Penerapan *Sniffer*

Menangkap atau mengendus (*sniffing*) trafik jaringan sangat berguna bagi administrator untuk menyelesaikan masalah-masalah yang muncul di jaringan, termasuk masalah keamanan. Hal ini berdasarkan fakta bahwa sejumlah penyerangan dimulai dengan penggunaan penyadap untuk melihat trafik jaringan dengan harapan melihat data-data penting yang ditransmisikan oleh *user*. Karena itu mempelajari kapabilitas dan keterbatasan *software-software packet sniffing* menjadi bagian yang penting dari kemampuan admin. *Software Open Source Sniffing* yang digunakan untuk melakukan *sniffing* adalah :

1. *TCPdum* adalah tools *UNIX/Linux* yang banyak digunakan untuk merekam trafik jaringan berdasarkan kriteria user dan mampu menyimpan trafik dalam berbagai format yang berbeda.
2. *Wireshark* adalah aplikasi grafis yang banyak digunakan untuk monitoring dan analisis. *Wireshark* adalah *opensource* dan berjalan pada banyak *platform OS* termasuk *Windows, Linux* dan *UNIX*.



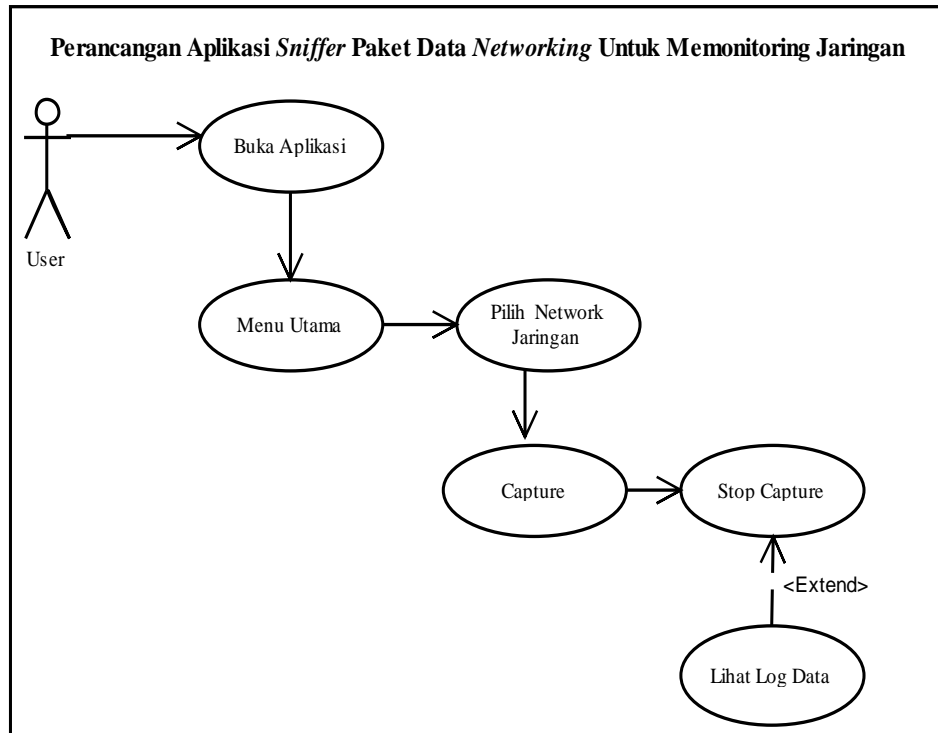
Gambar.III.1. Blok Diagram Paket

III.5. Desain Sistem

Pada perancangan aplikasi *sniffing* paket data *networking* untuk memonitoring jaringan ini akan dijelaskan mengenai rancangan aplikasi yang akan dikerjakan serta fitur-fitur yang akan dipakai pada aplikasi tersebut seperti algoritma perancangan, rancangan layar, dan lainnya.

III.6. Use Case Diagram

Use case diagram berfungsi untuk menggambarkan kegiatan aktor atau pengguna aplikasi, adapun *use case* diagram aplikasi yang dirancang dapat dilihat pada gambar III.2 berikut.

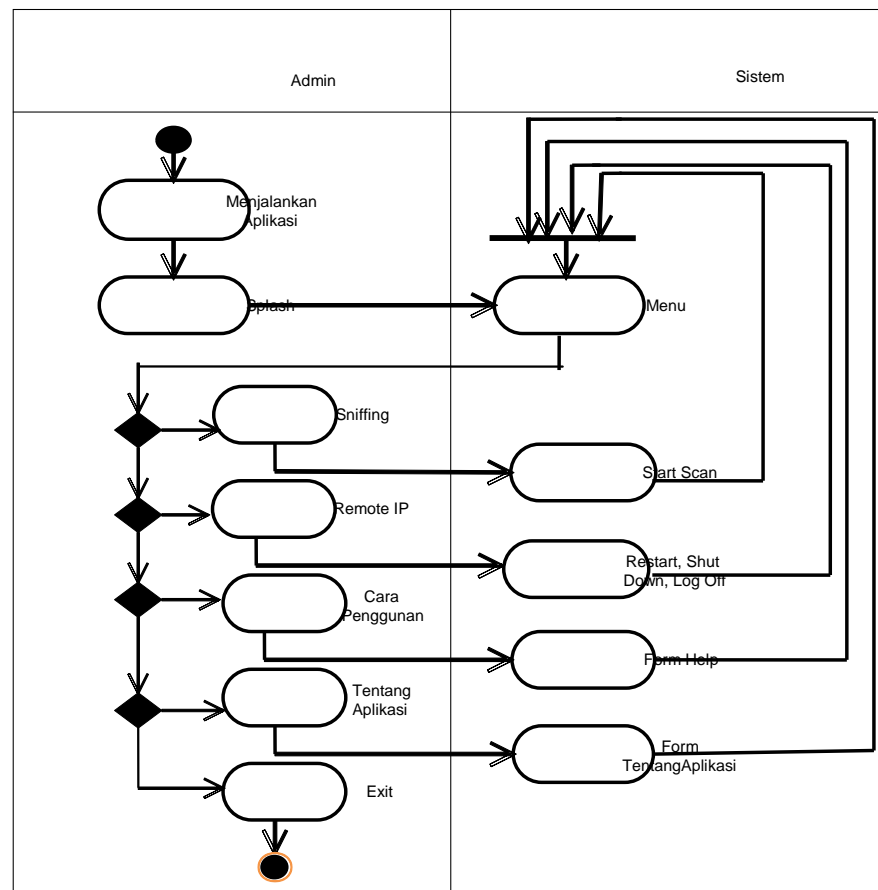


Gambar.III.2. Use Case Diagram

Dari gambar *usecase* diagram diatas, pengguna memulai aplikasi dan memilih menu *ip port scan*.

III.7. Activity Diagram

Pada *activity* diagram dibawah ini menggambarkan proses yang berjalan pada aplikasi *android* terdapat beberapa menu yang ditampilkan. Proses yang berlangsung terjadi setelah pengguna menjalankan aplikasi, yang dapat dilihat pada gambar III.3 berikut.



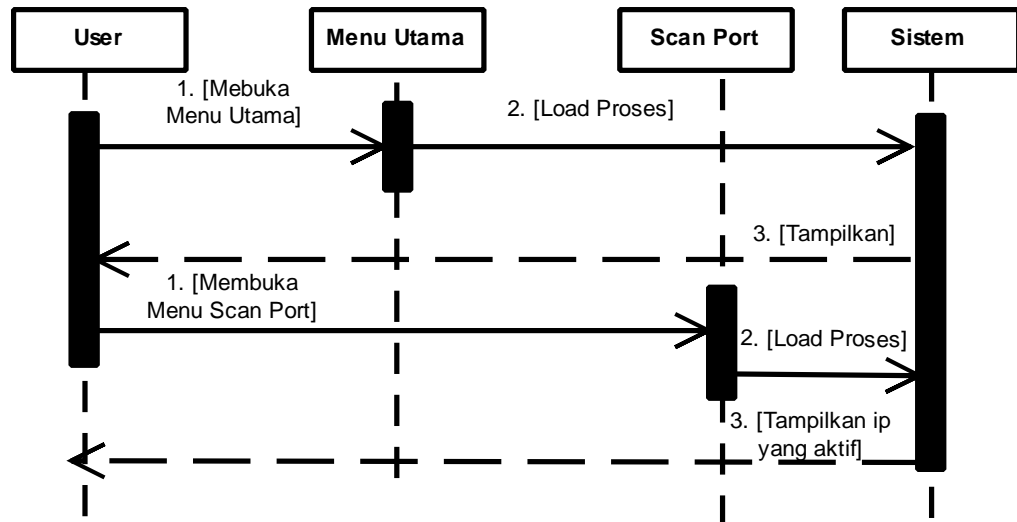
Gambar.III.3. Activity Diagram

Dari gambar *Activity* diagram diatas, proses aplikasi merupakan tahapan yang disajikan terhadap cara kerja aplikasi *sniffing* paket data ketika digunakan oleh pengguna.

III.8. *Sequence* Diagram

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

1. *Sequence Diagram Start Scan IP*, untuk diagram proses *start scan Ip* dapat dilihat pada gambar III.5 dibawah ini.



Gambar.III.4. Squence Diagram Scan Port

III.9. Perancangan *User Interface*

Aplikasi ini dibangun dengan desain *User interface form* atau tampilan pengguna yaitu *form* utama, *form help* dan *form about*. Berikut merupakan penjelasan dari desain *interface* yang dirancang.

III.9.1. Desain *Form* Menu Utama

Desain *form* Menu Utama merupakan tampilan untuk memproses *scan ip* monitoring, yang dapat dilihat pada gambar III.5 berikut ini.

Network Scanner	Paket Sniffing	

Gambar.III.5. Rancangan Menu Utama

III.9.2. Desain *Form* Proses Monitoring

Adapaun proses monitoring untuk menscan ip yang aktif pada jaringan

Wifi dapat dilihat pada gambar III.6 berikut ini:

Network Scanner	Package Sniffer		Filter
Network Device	Type	Hardware	Protocol

Gambar.III.6. Rancangan Menu *Proses Sniffer*