

BAB III

METODOLOGI PENELITIAN

Bab ini membahas tentang metodologi penelitian dalam melakukan perhitungan dan analisis kinerja level komponen dan protokol jaringan pada server di PT Telkom Divre V. Bahan penelitian, cara pengambilan data, jenis data, analisis sistem dan struktur tabel di bahas pada bab ini.

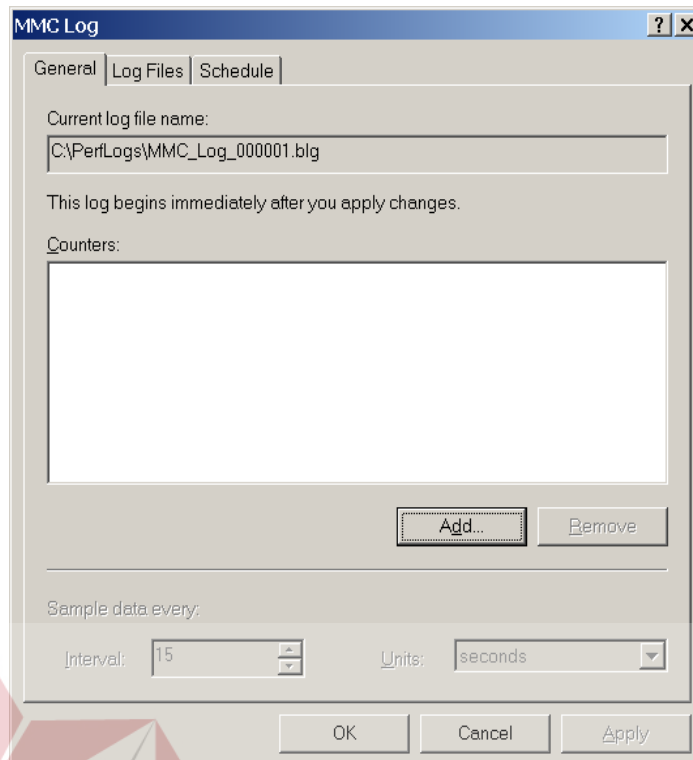
3.1 Bahan Penelitian

Sebagai bahan penelitian dalam analisis level komponen digunakan data dari file log Microsoft Management Console (MMC). Sedangkan file log Ethereal digunakan sebagai bahan penelitian untuk menganalisis protokol jaringan komputer.

3.1.1 Pengambilan Data

Pengambilan data dilakukan dengan melakukan *capture* pada server yang diamati, menggunakan MMC dan Ethereal. Data dari MMC berupa data atribut level komponen server meliputi physical disk, processor/CPU, sistem, memory dan Network Interface Card (NIC). Data file log Ethereal adalah data atribut network dengan menggunakan standar winpcap/tcpdump yang mengambil data pada OSI Layer 2 berupa frame.

File log dari MMC dapat diambil dengan cara melakukan *setting performance object* dan *counter* pada tool MMC. Dari *performance object* dapat dipilih obyek apa yang akan diamati, atribut serta *instance* apa saja dari tiap atribut.

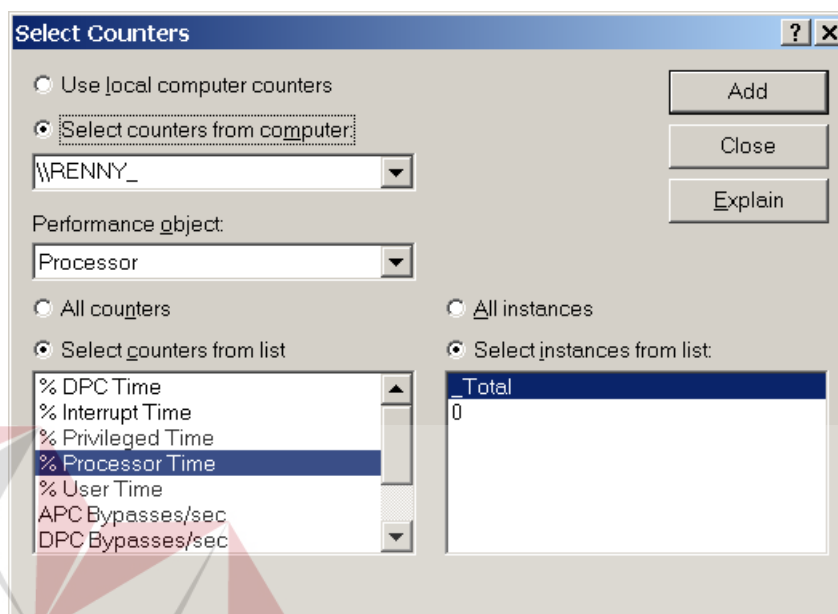


Gambar 3.1 Setting untuk membuat File Log

Pada menu MMC Log, seperti pada Gambar 3.1 dapat dilakukan setting nama file yang akan digunakan sebagai tempat penyimpanan file log. Interval atau selang waktu pengambilan capture data dapat diatur pada menu tersebut, pada penelitian ini digunakan interval terkecil yang disediakan pada MMC yaitu, setiap 1 detik. Hal ini disesuaikan dengan file log dari Ethereal yang rata-rata setiap 1 detik dapat melakukan capture sebanyak lebih dari 3 frame. Menu log MMC juga dapat digunakan untuk mengatur jadwal capture misalnya, dilakukan setiap tanggal tertentu setiap bulan selama waktu tertentu. Hasil file log dapat ditentukan format penyimpanannya, pilihan yang disediakan MMC adalah csv, .txt, .tsv dan Binary file.

Gambar 3.2 adalah Menu Select Counters, yang digunakan untuk melakukan *setting performance object, counter* dan *instance* pada MMC. Capture

dari komputer lain dapat dilakukan dengan memilih pada *field* Select counters from computer. Jenis object dapat diatur melalui *field* Performance object.



Gambar 3.2 Setting object, counter dan instance pada MMC

Data yang digunakan sebagai bahan penelitian meliputi physical disk, processor, Network Interface Card (NIC), memory, process dan sistem yang dapat dipilih melalui Performance object. Data detail *performance object* dan *counter* yang digunakan pada penelitian ini ditunjukkan pada Tabel 3.1.

Tabel 3.1 Tabel *Performance Object* yang digunakan

No	Performance Object	Counters
1	Processor	%DPC Time, %Interrupt Time, %Privileged Time, %Processor Time, %Use Time, Interrupt/sec.
2	Physical Disk	%Disk Read Time, %Disk Time, %Disk Write

		Time, %Idle Time, Disk Bytes/sec, Disk Read Bytes/sec, Disk Read/sec, Disk Transfers/sec, Disk Write Bytes/sec, Disk Writes/sec.
3	System	Processes, Threads, Processor Queue Length, File Write Bytes/sec, File Write Operations/sec, File Read Bytes/sec, File Read Operation/sec.
4	Memory	Page Faults/sec, Page Reads/sec, Page Writes/sec, Page Input/sec, Page Output/sec, Pages/sec.
5	Process	Creating Process ID, Elapsed Time, Handle Count, Thread Count.
6	Network_Interface	Bytes Received/sec, Bytes Sent/sec, Bytes Total/sec, Packets Received/sec, Packets Sent/sec.

Contoh capture data dari file log MMC seperti ditunjukkan pada Gambar 3.3.

```

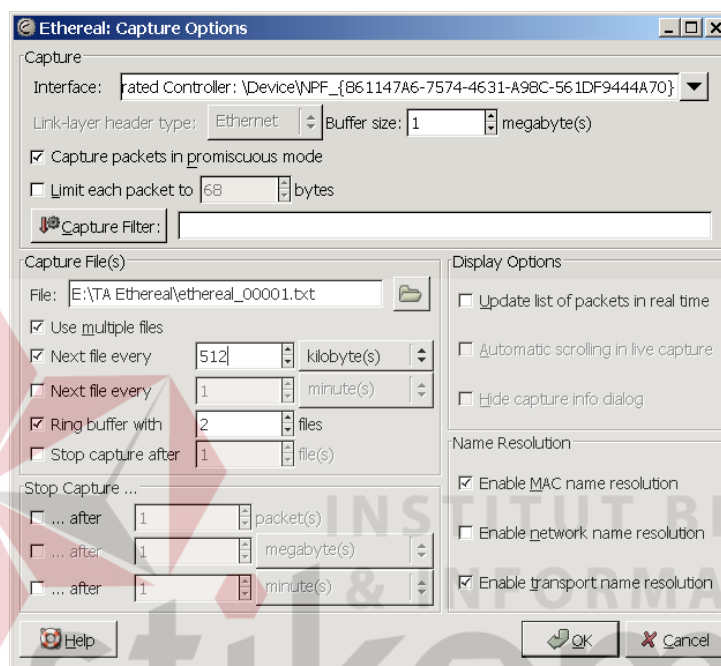
"(PDH-CSV 4.0) (SE Asia Standard Time)(-420)","%DPC
Time","%InterruptTime","%Privileged Time","%Processor Time","%User
Time",.....
"12/03/2004 08:58:56.625","6,472088850993852e-007","3,4100392824
268457e-005","99,990824001893358","3,0409622550157475e-005",.....

```

Gambar 3.3. Contoh file log dari MMC

Pada Gambar 3.3 baris pertama menunjukkan header *performance object* dan counter dari level komponen pada server, contoh %DPC Time, %Interrupt

Time pada komponen processor. Kemudian baris kedua adalah data performance dan counter, waktu capture data dilakukan pertama kali adalah pukul 08:58:56.625, tanggal 12/03/2004, DPC Time adalah 6,472088850993852e-007 dan seterusnya.



Gambar 3.4 Setting capture pada Ethereal

Data penelitian yang digunakan untuk menganalisis protokol jaringan diambil dari file log ethereal. Setting capture pada ethereal dapat dilihat pada Gambar 3.4. Dari gambar tampak bahwa capture dari ethereal akan membuat file baru jika ukuran file sebelumnya sudah mencapai 512 Kbytes pada *field* Next file every dan file hasil capture disimpan pada directori E:\TA\Ethereal\ethereal_00001.txt. Dari file log hasil capture tersebut akan dipilih beberapa protokol, yaitu : UDP, TCP, EIGRP, ARP, STP dan HTTP. Data detail file log Ethereal ditunjukkan pada Tabel 3.2.

Tabel 3.2 Data dari file log Ethereal

No	Data dari Ethereal	Keterangan
1	Frame	No Frame yang di capture oleh Ethereal.
2	Arrival Time	Waktu kedatangan Frame di capture.
3	Time delta from previous packets	Selang waktu antara Frame ke n dengan Frame n-1.
4	Time since reference or first frame	Jarak waktu antara Frame ke n dengan Frame pertama.
5	Packet Length	Panjang packet dalam satuan bytes.
6	Capture Length	Panjang packet yang di capture dalam satuan bytes.
7	Destination MAC	Alamat MAC Address tujuan dimana Frame dikirim.
8	Source MAC	Alamat MAC Address asal Frame di kirim.
9	Protocol	Jenis protokol dari Frame.
10	Destination IP	Alamat IP Address tujuan Frame dikirim.
11	Source IP	Alamat IP Address asal Frame dikirim.
12	Destination Port	Alamat Port tujuan Frame dikirim.
13	Source Port	Alamat Port asal Frame dikirim.
14	Seq	Sequence Number pengiriman Frame
15	Ack	Acknowledge pengiriman Frame

File log Ethereal berformat tcpdump tersimpan secara binary sehingga tidak dapat dibaca dengan text editor biasa. Untuk memudahkan pembacaan dengan text editor, Ethereal menyediakan fungsi untuk mengekspor/memindahkan dari format tcpdump ke plain text atau ke format yang lain.

Contoh data dari log Ethereal dengan format plain text seperti ditunjukkan pada Gambar 3.5.

```

Frame 1 (60 bytes on wire, 60 bytes captured)
  Arrival Time: Dec 3, 2004 08:56:34.916218000
  Time delta from previous packet: 0.000000000 seconds
  Time since reference or first frame: 0.000000000 seconds
  Frame Number: 1
  Packet Length: 60 bytes
  Capture Length: 60 bytes

IEEE 802.3 Ethernet
  Destination:01:80:c2:00:00:00 (Spanning-tree-(for-bridges)_00)
  Source: 00:0b:46:98:36:12 (Cisco_98:36:12)
  Length: 38
  Trailer: 0000000000000000
  Logical-Link Control
  Spanning Tree Protocol

```

Gambar 3.5. Contoh file log Ethereal

Dari Gambar 3.5 dapat dilihat detail hasil capture dari Ethereal, Frame 1 (60 bytes on wire, 60 bytes captured) menunjukkan urutan frame yang tercapture pada suatu file adalah pada urutan no 1, kemudian besar frame adalah 60 bytes. Arrival Time: Dec 3, 2004 08:56:34.916218000 adalah waktu dilakukan capture frame tersebut. Time delta from previous packet: 0.000000000 seconds menunjukkan bahwa selang waktu dengan frame yang sebelumnya adalah 0,000000000 second. Time since reference or first frame: 0.000000000 seconds

adalah selang waktu dengan kedatangan frame pertama. Informasi lain adalah Destination MAC Address : 01:80:c2:00:00:00, Source MAC Address : 00:0b:46:98:36:12, Protokol : Spanning Tree Protocol (STP).

Data file log Ethereal dan MMC selanjutnya akan dimasukkan ke database MySQL dengan bantuan interface. Interface ini akan membaca file dengan format plain text dan csv untuk dipindahkan ke tabel-tabel yang sudah disediakan.

3.1.2 Variabel penelitian

Metode yang digunakan untuk menghitung dan menganalisis data adalah *Performance Modelling Concepts*. Beberapa variabel yang diperlukan untuk menerapkan metode ini adalah (Bobbin, 2004):

- a. Periode waktu selama pengamatan dilakukan, diidentifikasi dengan **T**.
- b. Jumlah task yang datang ke server selama rentang waktu tertentu **T**, diidentifikasi dengan **A**.
- c. Jumlah task yang dapat diselesaikan dalam rentang waktu tertentu **T**, diidentifikasi dengan **C**.
- d. Jumlah total waiting time, diidentifikasi dengan **W**.
- e. Jumlah transaksi yang dapat diselesaikan oleh *server* **C(0)**.
- f. Total Frame yang dapat di-*capture*.
- g. Rata-rata packet size per frame.
- h. Rata-rata time per bytes.
- i. Rata-rata bytes per time.

Data tersebut di atas akan digunakan sebagai input untuk menghitung kinerja level komponen dan protokol jaringan pada server. Perhitungan yang dilakukan berdasarkan pada *Performance Modelling Concepts*, yang pernah dilakukan pada penelitian terdahulu.

3.1 Analisis Sistem

Sistem yang dibuat adalah sistem yang dapat digunakan untuk menghitung dan menganalisis kinerja level komponen dan protokol jaringan komputer. Sistem ini terdiri dari 2 buah interface untuk memindahkan file log ke dalam database MySQL serta 1 buah modul untuk melakukan perhitungan dan analisis data.

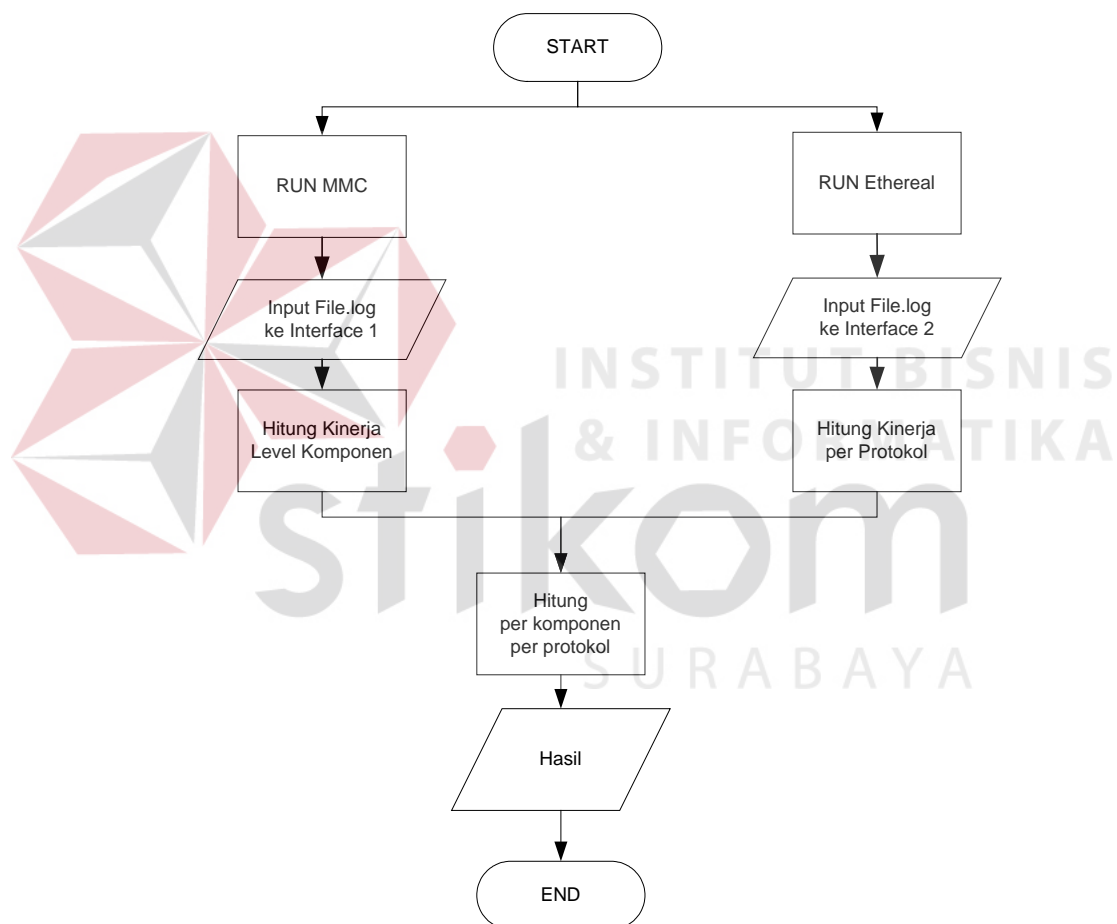
Interface 1 digunakan untuk membaca file log MMC kemudian memindahkannya ke database dalam bentuk tabel, sehingga perhitungan lebih mudah dilakukan. Interface 1 akan memindahkan data ke tabel processor, process, sistem, memory, disk dan NIC. Tabel-tabel ini digunakan untuk melakukan perhitungan dengan menggunakan *Performance Modelling Concepts*.

Interface 2 digunakan untuk membaca file log Ethereal dan memindahkannya ke database dalam bentuk tabel. Tabel yang akan dipakai oleh interface ini adalah Tabel Frame. Tabel ini digunakan untuk menghitung nilai rata-rata dari frame size, waktu dan hal-hal lain yang berhubungan dengan detail frame yang tercapture.

Satu modul lagi dibuat sebagai interface untuk memudahkan user dalam menggunakan sistem ini. Rekap hasil perhitungan dan analisis level komponen berada pada modul ini. Dalam rekap yang dibuat dihitung break down perhitungan

per komponen per protokol, untuk komponen processor, physical disk, incoming link, outgoing link serta protokol UDP, TCP, EIGRP, ARP, STP dan HTTP.

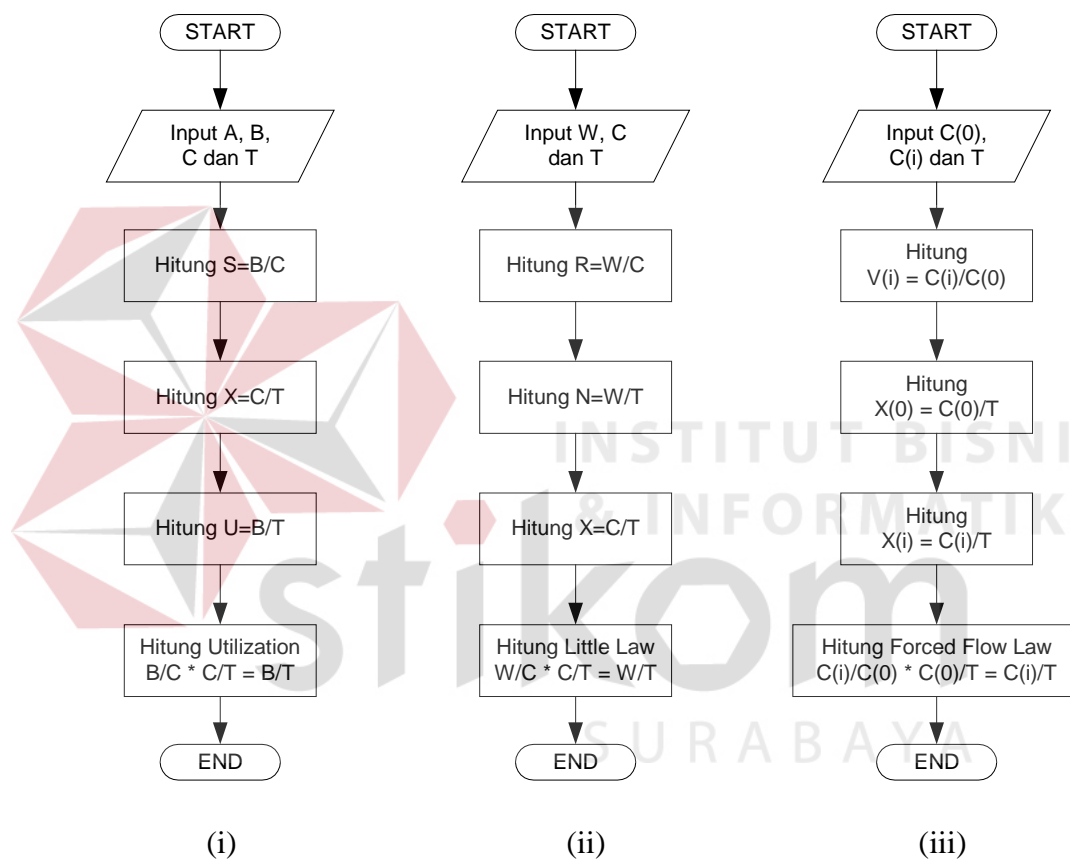
Gambar 3.6. menunjukkan flowchart atau diagram alir dari sistem analisis kinerja level komponen dan protokol jaringan yang akan dibuat. Beberapa proses yang dominan adalah Hitung Kinerja Level Komponen, Hitung Kinerja per Protokol dan Hitung per Komponen per Protokol.



Gambar 3.6 Flowchart Analisis Kinerja Level Komponen dan Protokol

Hitung Kinerja Level Komponen terdiri dari 3 sub proses dengan menggunakan hukum Little's Law, Forced Flow Law dan Utilization Law.

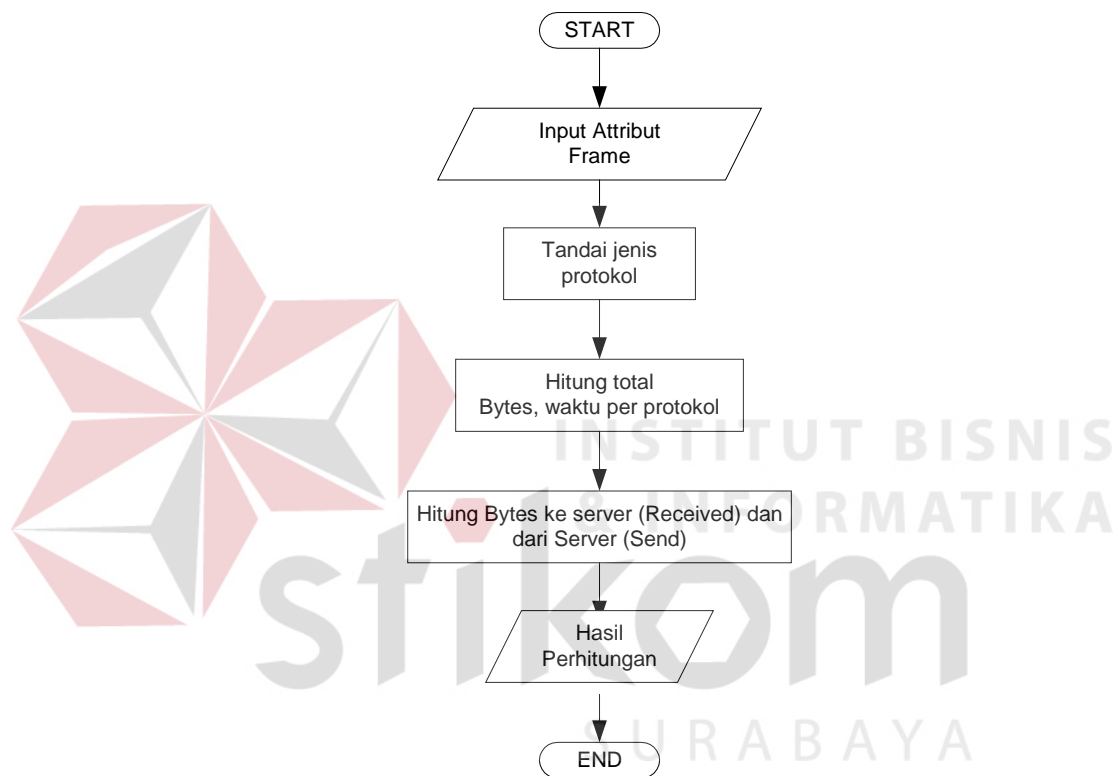
Gambar 3.7 menunjukkan flowchart untuk menghitung dan menganalisis dengan *Performance Modelling Concepts*. Tiap Flowchart akan menerima input dari sistem yang diambil dari database. Dari masing-masing input akan digunakan untuk melewati tahap-tahap perhitungan sampai didapatkan nilai Utilization, Little's Law dan Forced Flow Law.



Gambar 3.7 Flowchart Utilization, Little's Law dan Forced Flow Law

Proses Hitung Kinerja per protokol akan menghasilkan jumlah total frame, jumlah bytes, rata-rata waktu yang diperlukan untuk melewati 1 frame, rata-rata waktu yang diperlukan untuk melewati 1 bytes, rata-rata bytes dikirim per satuan waktu tertentu. Proses ini juga dapat mengetahui jumlah frame menurut protokol, sehingga dapat diketahui protokol paling dominan yang melewati server.

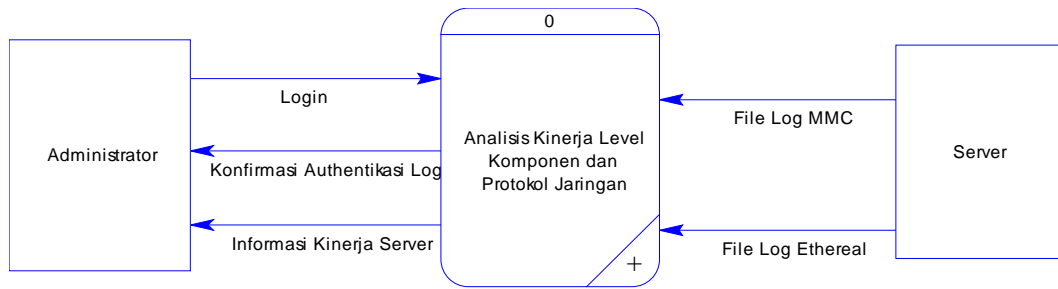
Proses Hitung Komponen per protokol memperoleh input dari file log Ethereal yang telah disimpan ke database, berupa ukuran frame dalam bytes, waktu frame melewati server, jenis protokol frame, source dan destination MAC Address, IP Address dari mana dan ke mana frame dikirim. Tahap-tahap perhitungan analisis per protokol ditunjukkan pada Gambar 3.8.



Gambar 3.8 Flowchart menghitung protokol

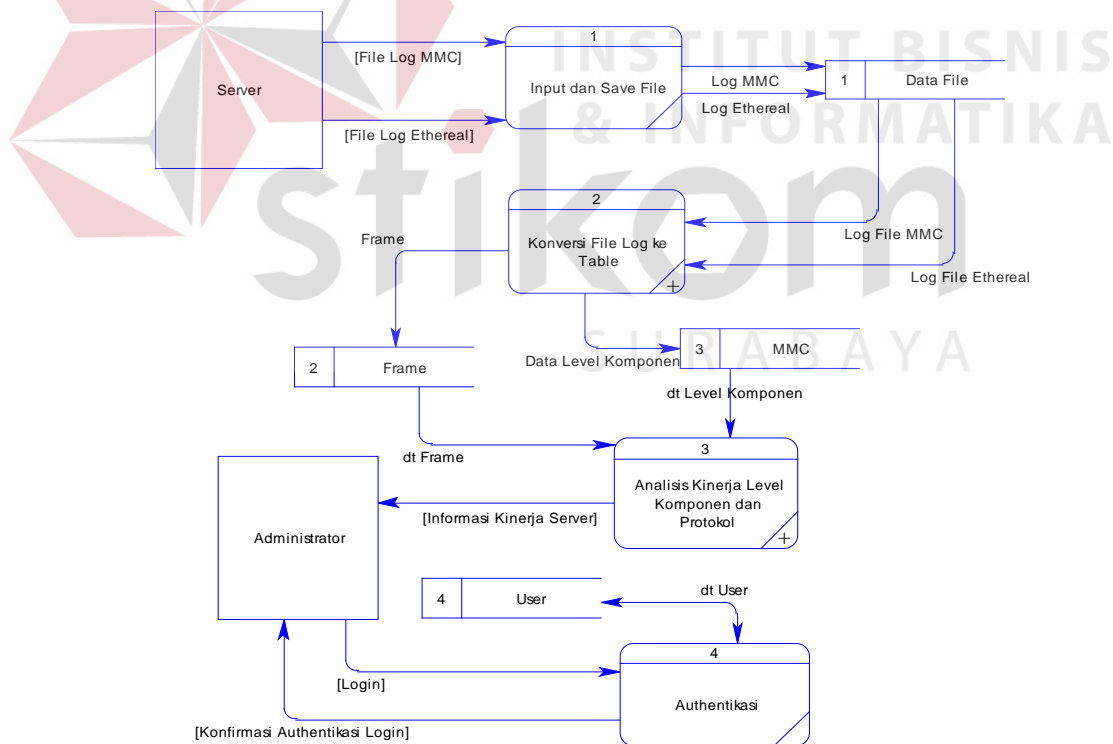
3.2 DFD

Dari analisis sistem di atas dapat disusun Data Flow Diagram (DFD) untuk membuat sistem analisis kinerja level komponen dan protokol jaringan. DFD ini menjelaskan tentang proses dan sub proses serta input dan output sistem secara detail.



Gambar 3.9 DFD level 0 sistem analisis kinerja level komponen dan protokol

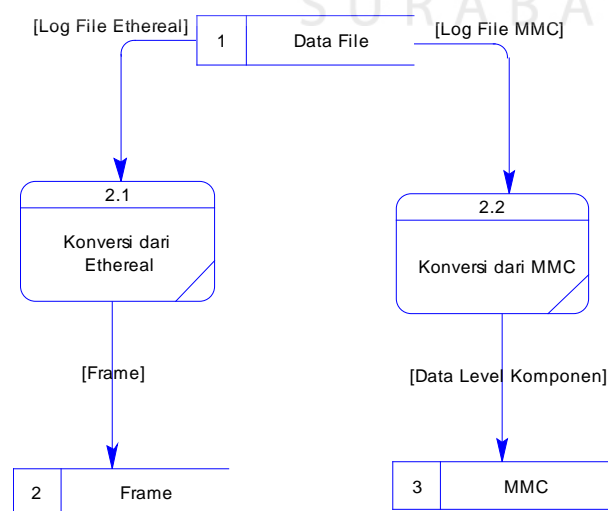
Pada Gambar 3.9 ditunjukkan bahwa sistem mendapat input berupa autentikasi login dan password dan log file ethereal dengan format plain text dari Administrator. Dari server, sistem mendapat input berupa hasil log file dari MMC dan Ethereal yang dijalankan di server. Administrator mendapat output dari sistem berupa informasi kinerja server.



Gambar 3.10 DFD Level 1 Analisis Kinerja Level Komponen dan Protokol

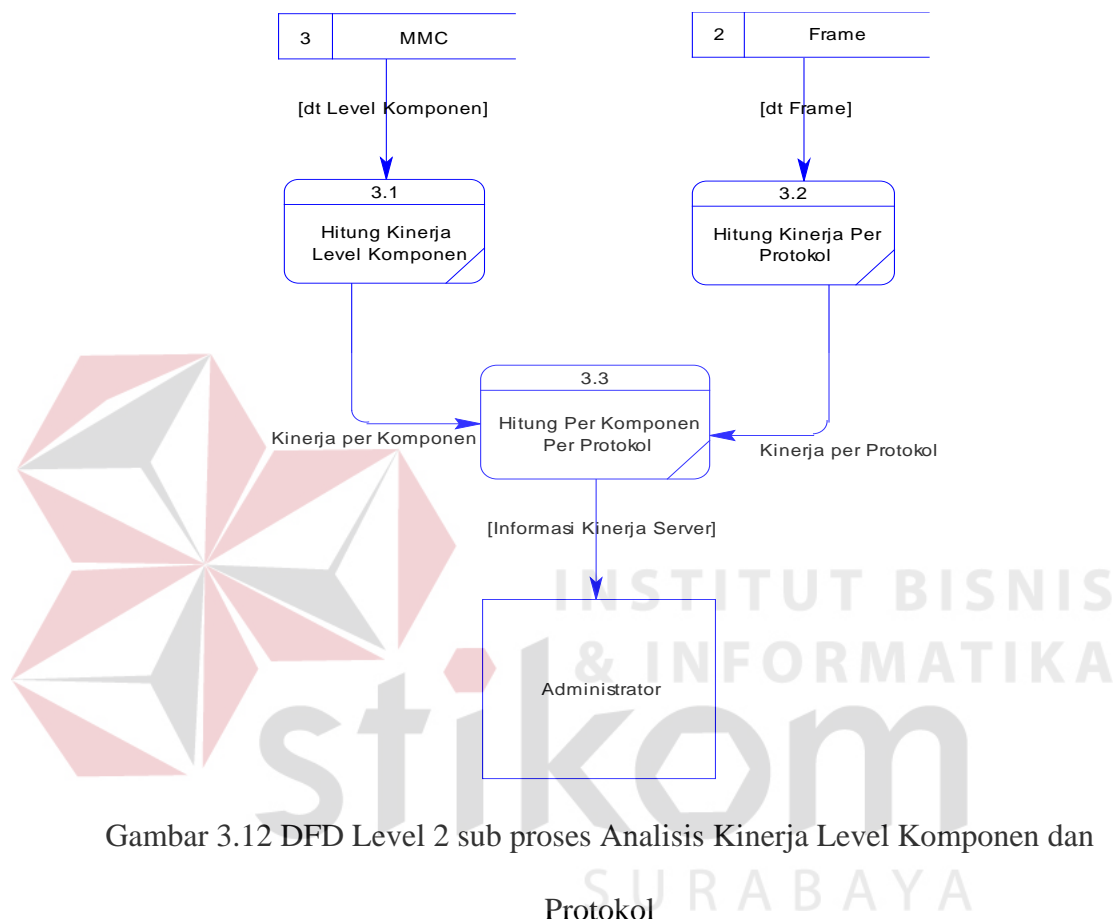
Pada Gambar 3.10 terlihat beberapa sub proses yaitu : Input dan Save File, Konversi File Log ke Tabel, Analisis Kinerja Level Komponen dan Authentikasi. Sub proses Input dan Save File mendapatkan input dari Administrator berupa File log dari MMC dengan format .csv serta file log Ethereal yang sudah di ekspor terlebih dahulu ke format plain teks/txt. File log asli akan di upload ke web server dan attribut filenya akan disimpan dalam tabel datafile.

Sub proses berikutnya adalah Konversi File Log ke Tabel yang mendapatkan masukan dari tabel datafile berupa attribut file yang akan dipindahkan ke tabel Frame dan tabel-tabel MMC. Tabel-tabel ini akan memberi masukan ke sub proses Analisis Kinerja Level Komponen dan Protokol sebagai dasar untuk melakukan perhitungan. Sub proses terakhir adalah Authentikasi user dan password untuk mengakses keseluruhan sistem yang berhubungan langsung dengan user yaitu Administrator. Sub proses ini akan mengakses tabel User pada saat authentikasi login dan password dilakukan.



Gambar 3.11 DFD Level 2 Sub proses Konversi File Log ke Tabel

Gambar 3.8 menunjukkan DFD level 2 dari sub proses Konversi File Log ke Tabel. Sub proses ini terdiri dari 2 proses yaitu : Konversi dari Ethereal dan MMC ke tabel-tabel yang telah ditentukan.



Gambar 3.12 DFD Level 2 sub proses Analisis Kinerja Level Komponen dan Protokol

Pada Gambar 3.12 tampak bahwa sub proses Analisis Kinerja Level Komponen dan Protokol mempunyai 3 proses yaitu : Hitung Kinerja Level Komponen, Hitung Kinerja per Protokol dan Hitung Per Komponen per Protokol. Proses Hitung Kinerja Level Komponen mendapat input dari tabel dan menghasilkan output berupa data hasil perhitungan kinerja per komponen. Detail dari proses ini ditunjukkan pada Gambar 3.7 dengan menggunakan 3 hukum dasar yaitu Little's Law, Forced Flow Law dan Utilization Law.

Proses Hitung Kinerja per protokol mendapat input dari tabel Frame dan menghasilkan output berupa data hasil perhitungan per protokol. Detail proses secara *pseudocode* adalah sebagai berikut :

- a. Hitung jumlah frame.
- b. Hitung jumlah total bytes seluruh frame.
- c. Hitung selang waktu antara frame pertama dan terakhir.
- d. Hitung panjang packet yang menuju server dan dari server.
- e. Tandai frame dengan protokol yang sama.
- f. Hitung rata-rata packet size/frame, rata-rata time/frame, rata-rata time/bytes dan rata-rata bytes/time untuk masing-masing protokol.

3.3 Struktur Tabel

- a. Tabel Datafile

Primary Key : Id_datafile

Fungsi : untuk menyimpan attribut file log serta file dalam web server

Nama	Tipe	Ukuran	Deskripsi
Id_datafile	Bigint	5	Id untuk tabel Datafile
Nama	Varchar	25	Nama file .log
Size	Bigint	10	Ukuran file .log
Type	Varchar	15	Tipe file .log
IP_Server	Varchar	15	IP Server yang akan dianalisis
Nama_database	Varchar	15	Nama Database di mana tabel datafile disimpan
Jenis_log	Varchar	10	Jenis log file

b. Tabel Disk

Primary Key : Id_Disk

Fungsi : untuk menyimpan atribut disk dari MMC

Nama	Tipe	Ukuran	Deskripsi
Id_disk	Bigint	15	Id untuk tabel Disk
Standart_Time	Varchar	30	Tanggal dan waktu dilakukan
Disk_read_time	Varchar	10	Waktu disk dalam keadaan read
Disk_time	Varchar	10	Waktu disk
Disk_write_time	Varchar	10	Waktu disk saat write
Idle time	Varchar	10	Waktu disk dalam keadaan idle
Avg_disk_sec_read	Varchar	10	Rata-rata waktu disk read
Avg_disk_sec_write	Varchar	10	Rata-rata waktu disk write
Disk_bytes_sec	Varchar	10	Proses pada disk dengan satuan bytes/sec
Avg_disk_queue_length	Varchar	10	Rata-rata panjang antrian
Avg_disk_bytes_read	Varchar	10	Rata-rata melakukan read per byte
Avg_disk_bytes_write	Varchar	10	Rata-rata melakukan write per byte

c. Tabel Frame

Primary Key : Id

Fungsi : untuk menyimpan data Frame hasil capture dari Ethereal

Nama	Tipe	Ukuran	Deskripsi
Id	Bigint	15	Id untuk Frame
Frame_Number	Bigint	15	Nomor Frame
Arrival_Time	Varchar	20	Waktu Frame di-capture
Date	Varchar	20	Tanggal Frame di-capture
Protocol	Varchar	8	Jenis Protokol dari Frame

Nama	Tipe	Ukuran	Deskripsi
Packet_Length	Varchar	5	Panjang packet
Time_delta_from_previous_packet	Varchar	15	Selang waktu Frame n dengan Frame n-1/sebelumnya
Time_since_ref_or_first_packet	Varchar	15	Selang waktu Frame n dengan Frame pertama
Length	Varchar	11	Panjang frame
IP_Source	Varchar	15	IP source
IP_Destination	Varchar	15	IP destination
MAC_Source	Varchar	14	MAC Address Source
MAC_Destination	Varchar	17	MAC Address Destination

d. Tabel Memory

Primary Key : Id_memory

Fungsi : untuk menyimpan data memory dari server

Nama	Tipe	Ukuran	Deskripsi
Id_memory	Bigint	15	Id untuk tabel Memory
Page Faults/sec	Varchar	10	Page Faults/sec
Page Read/sec	Varchar	10	Page Read/sec
Page Write/sec	Varchar	10	Page Write/sec
Page Input/sec	Varchar	10	Page Input/sec
Page Output/sec	Varchar	10	Page Output/sec
Pages/sec	Varchar	10	Pages/sec

e. Tabel Network_Interface

Primary Key : Id_nic

Fungsi : untuk menyimpan data yang dapat disimpan dari attribut NIC server

Nama	Tipe	Ukuran	Deskripsi
Id_nic	Bigint	15	Id untuk tabel Network Interface
Bytes Received/sec	Varchar	30	Bytes Received/sec
Bytes Sent/sec	Varchar	30	Bytes Sent/sec
Bytes Total/sec	Varchar	30	Bytes Total/sec
Packets Received/sec	Varchar	30	Packets Received/sec
Packets Sent/sec	Varchar	30	Packets Sent/sec
Current Bandwidth	Varchar	30	Current Bandwidth

f. Tabel Process

Primary Key : Id_process

Fungsi : untuk menyimpan data process dari server

Nama	Tipe	Ukuran	Deskripsi
Id_process	Bigint	5	Id untuk tabel process
Creating Process ID	Varchar	15	Creating Process ID
Elapsed Time	Varchar	15	Elapsed Time
Handle Count	Varchar	15	Handle Count
Thread Count	Varchar	15	Thread Count
User Time	Varchar	15	User Time
Page Faults/sec	Varchar	15	Page Faults/sec

g. Tabel Processor

Primary Key : Id_processor

Fungsi : untuk menyimpan data processor dari server

Nama	Tipe	Ukuran	Deskripsi
Id_processor	Bigint	5	Id untuk tabel processor
DPC Time	Varchar	25	DPC Time
Interrupt Time	Varchar	10	Interrupt Time
Privileged Time	Varchar	15	Privileged Time
Processor Time	Varchar	15	Processor Time
Use Time	Varchar	20	Use Time
Interrup/sec	Varchar	20	Interrup/sec

h. Tabel Sistem

Primary Key : Id_sistem

Fungsi : untuk menyimpan data sistem dari server

Nama	Tipe	Ukuran	Deskripsi
Id_sistem	Bigint	5	Id untuk tabel sistem
Processes	Varchar	25	Processes
Threads	Varchar	25	Threads
Processor Queue Length	Varchar	25	Processor Queue Length
File Write Bytes/sec	Varchar	25	File Write Bytes/sec
File Write Operations/sec	Varchar	25	File Write Operations/sec
File Read Operations/sec	Varchar	25	File Read Operations/sec

i. Tabel Username

Primary Key : Id_user

Fungsi : untuk menyimpan username yang dapat melakukan akses

Nama	Tipe	Ukuran	Deskripsi
Id_user	Bigint	5	Id untuk username
Username	Varchar	25	Username

Nama	Tipe	Ukuran	Deskripsi
Password	Varchar	10	Password

