

# Pengembangan Model dan Analisa Routing Protocol Menggunakan OPNET Modeler 14.0

Dwi Haryono

*Jurusan Teknik Informatika, STMIK-AMIK Riau*

*dwharyono\_skom@yahoo.com*

## **Abstrak**

*Penelitian ini dilakukan untuk merancang dan menganalisa sebuah Routing Protocol yang dibuat dengan Software OPNET Modeler 14.0 yang diimplementasikan di STMIK AMIK Riau. Dalam memperoleh data tim lakukan dengan melakukan observasi, membaca buku dan literature. Dengan adanya sistem ini dapat membantu kampus untuk penerapan teknologi jaringan berbasis LAN dalam mengimplementasikan sistem yang akan diterapkan dengan menggunakan basis routing protocol yang tercepat. Untuk mengetahui routing tercepat atau terpendek dengan membandingkan routing protocol OSPF dan RIP dalam jaringan disimulasikan dalam software OPNET Modeler 14.0, setelah itu akan didapatkan hasil sesuai yang diharapkan.*

*Kata kunci : LAN, Routing Protocol, OSPF dan RIP, software OPNET Modeler 14.0*

## **1. Pendahuluan**

### **1.1. Latar Belakang**

Menurut (Nassar & Daniel, 2000) & (Stalling & William, 1997) Pada era global seperti saat ini, informasi sudah merupakan salah satu kebutuhan dasar manusia. Akibatnya, pemakaian internet sebagai jaringan informasi global semakin meningkat. Kecepatan pertumbuhan jumlah pemakai jaringan global dan perkembangan internet telah menjadikan lalu lintas internet menjadi ramai. Oleh sebab itu, suatu router harus mampu untuk melakukan proses routing untuk menentukan jalan tercepat atau terdekat dalam mengalirkan paket-paket data agar sampai ke tujuannya. Aturan router dalam melakukan proses

routing tersebut dikenal dengan protokol routing. Protokol routing yang ada yaitu Routing Information Protocol (RIP). perkembangan jaringan IP (Internet Protokol) yang semakin besar dan Internet Engineering Task Force (IETF) mengembangkan suatu protokol routing untuk mengatasi kekurangan pada RIP yaitu Open Shortest path First atau disingkat OSPF. OSPF bekerja berdasarkan algoritma Shortest Path First (SPF) yang dikembangkan berdasarkan algoritma Dijkstra. OSPF mendistribusikan informasi routingnya di dalam router-router yang tergabung ke dalam suatu Autonomous System (AS).

OSPF merupakan sebuah routing protokol berjenis IGP (Interior Gateway Protocol) yang hanya dapat bekerja dalam jaringan internal suatu organisasi atau perusahaan. Jaringan internal maksudnya adalah jaringan di mana kita masih memiliki hak untuk menggunakan, mengatur, dan memodifikasinya. Atau dengan kata lain, kita masih memiliki hak administrasi terhadap jaringan tersebut. Jika kita sudah tidak memiliki hak untuk menggunakan dan mengaturnya, maka jaringan tersebut dapat dikategorikan sebagai jaringan eksternal.

OSPF merupakan routing protokol yang menggunakan konsep hirarki routing, artinya OSPF membagi-bagi jaringan menjadi beberapa tingkatan. Tingkatan ini diwujudkan dengan menggunakan sistem pengelompokan area. Dengan menggunakan konsep hirarki routing ini sistem penyebaran informasinya menjadi lebih teratur dan tersegmentasi.

### **1.2. Rumusan Masalah**

Rumusan Masalah dalam Penelitian ini adalah menganalisa “Bagaimana kinerja *Routing protocol* dalam menyalurkan data dengan menggunakan *Ethernet* sebagai *input* nilai *Load (bits/sec)* dengan

menggunakan perbandingan *routing protocol OSPF dan RIP*”.

### 1.3. Batasan Masalah

Dalam timan Penelitian ini tim hanya membatasi :

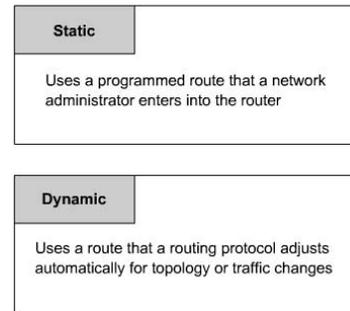
- Menganalisa kinerja *Routing protocol* dalam menyalurkan data.
- Membandingkan *OSPF dan RIP*.
- Dalam implementasi tim menggunakan simulasi OPNET Modeler 14.0

## 2. Landasan Teori

Pengiriman dan melewati informasi atau data tidak lepas dari *topologi* besar yang menghubungkan banyak jaringan dengan bermacam *segmen*, hal ini terbantu dengan adanya *router* yang dapat melewati *protocol* walau berbeda, namun dengan banyaknya *system routing protocol* yang ada sekarang memiliki perbedaan *performance* dalam melewati data, hal ini yang menjadi inspirasi tim untuk menganalisa dengan merancang jaringan dalam bentuk simulasi menggunakan OPNET Modeler 14.0.

### 2.1 Routing

Menurut (Downes & Kevin, 1998), Routing adalah sebuah proses untuk meneruskan paket jaringan dari satu jaringan ke jaringan lainnya melalui sebuah *internetwork*. *Routing* juga dapat merujuk kepada sebuah metode penggabungan beberapa jaringan sehingga paket-paket data dapat hinggap dari satu jaringan ke jaringan selanjutnya. Untuk melakukan hal ini, digunakanlah sebuah perangkat jaringan yang disebut sebagai *router*. *Router* tersebut akan menerima paket yang ditujukan ke jaringan di luar jaringan yang pertama, dan akan meneruskan paket yang ia terima kepada *router* lainnya hingga sampai kepada tujuannya.



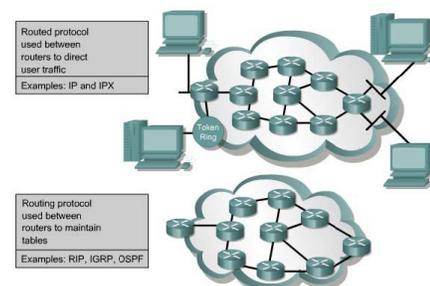
Gambar 1. Type Routing

*Routing* dapat di bedakan berdasarkan Konfigurasi menjadi 2 yaitu :

- Routing Static*, Konfigurasinya harus dilakukan secara manual oleh *administrator* jaringan harus memasukkan atau menghapus *route statis* jika terjadi perubahan *topologi*. Pada jaringan skala besar, jika tetap menggunakan *routing statis*, maka akan sangat membuang waktu *administrator* jaringan untuk melakukan *update table routing*. jadi *routing statis* hanya mungkin dilakukan untuk jaringan skala kecil.
- Routing Dynamic*, *routing dinamis* bisa diterapkan di jaringan skala besar dan membutuhkan kemampuan lebih dari administrator. *Routing protocol* adalah berbeda dengan *routed protocol*. *Routing protocol* adalah komunikasi antara *router*. *Routing protocol* mengijinkan *router-router* untuk *sharing* informasi tentang jaringan dan koneksi antar *router*. *Router* menggunakan informasi ini untuk membangun dan memperbaiki *table routingnya*.

Contoh *routing protocol*:

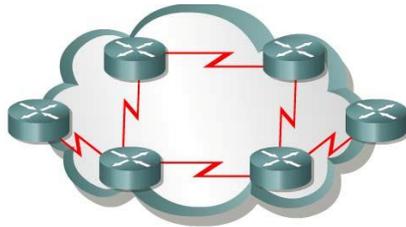
- Routing Information Protocol (RIP)
- Interior Gateway Routing Protocol (IGRP)
- Enhanced Interior Gateway Routing Protocol (EIGRP)
- Open Shortest Path First (OSPF)



Gambar 2. Routed vs Routing Protokol

### 2.2.1. Autonomous System (AS)

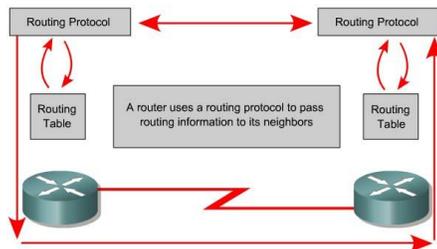
AS adalah kumpulan dari jaringan-jaringan yang dalam satu administrasi yang mempunyai strategi *routing* bersama. AS mungkin dijalankan oleh satu atau lebih operator ketika AS digunakan pada *routing* ke dunia luar.



Gambar 3. Autonomous System

Tujuan utama dari *routing* protokol adalah untuk membangun dan memperbaiki *table routing*. Dimana tabel ini berisi jaringan-jaringan dan *interface* yang berhubungan dengan jaringan tersebut. Router menggunakan *protokol routing* ini untuk mengatur informasi yang diterima dari *router-router* lain dan *interfacenya* masing-masing, sebagaimana yang terjadi di *konfigurasi routing* secara manual.

*Algoritma routing* adalah dasar dari *routing dinamis*. Kapanpun *topologi* jaringan berubah karena perkembangan jaringan, konfigurasi ulang atau terdapat masalah di jaringan, maka *router* akan mengetahui perubahan tersebut. Dasar pengetahuan ini dibutuhkan secara akurat untuk melihat *topologi* yang baru.



Gambar 4. Prinsip Kerja Autonomous System

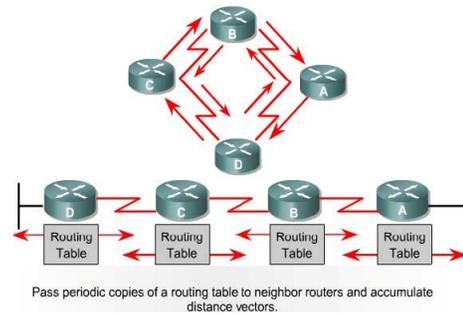
### 2.2.2. Klasifikasi Routing Protokol

Sebagian besar algoritma *routing* dapat di klasifikasikan menjadi satu dari dua kategori berikut :

#### 2.2.2.1. Routing Protokol Distance Vector

Algoritma *routing* distance vector secara periodik

menyalin *table routing* dari *router* ke *router*. Perubahan *table routing* ini *di-update* antar *router* yang saling berhubungan pada saat terjadi perubahan *topologi*. *Algoritma distance vector* juga disebut dengan *algoritma Bellman-Ford*. Setiap *router* menerima *table routing* dari *router* tetangga yang terhubung langsung. Pada gambar di bawah ini digambarkan konsep kerja dari *distance vector*.



Gambar 5. Konsep Distance Vector

Keterangan :

B menerima informasi dari Router A. Router B menambahkan nomor *distance vector*, seperti jumlah *hop*. Jumlah ini menambahkan *distance vector*. Router B melewatkan *table routing* baru ini ke *router-router* tetangganya yang lain, yaitu Router C. Proses ini akan terus berlangsung untuk semua *router*.

*Algoritma* ini mengakumulasi jarak jaringan sehingga dapat digunakan untuk memperbaiki *database* informasi mengenai *topologi* jaringan. Bagaimanapun, *algoritma distance vector* tidak mengijinkan *router* untuk mengetahui secara pasti *topologi internetwork* karena hanya melihat *router-router* tetangganya.

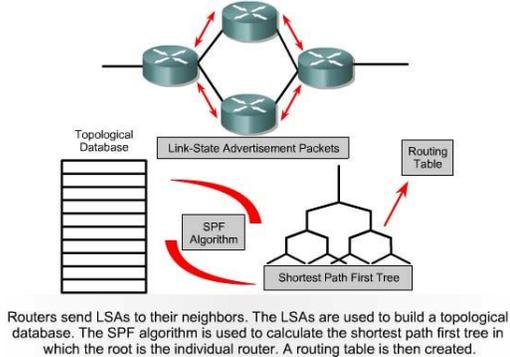
#### 2.2.2.2. Routing Protokol Link-state

*Algoritma link-state* juga dikenal dengan *algoritma Dijkstra* atau *algoritma shortest path first (SPF)*. *Algoritma* ini memperbaiki informasi *database* dari informasi *topologi*. *Algoritma distance vector* memiliki informasi yang tidak spesifik tentang *distance network* dan tidak mengetahui jarak *router*. Sedangkan *algoritma link-state* memperbaiki pengetahuan dari jarak *router* dan bagaimana mereka *inter-koneksi*.

Fitur-fitur yang dimiliki oleh *routing link-state* adalah:

- a. *Link-state advertisement (LSA)* – adalah paket kecil dari informasi *routing* yang dikirim antar *router*

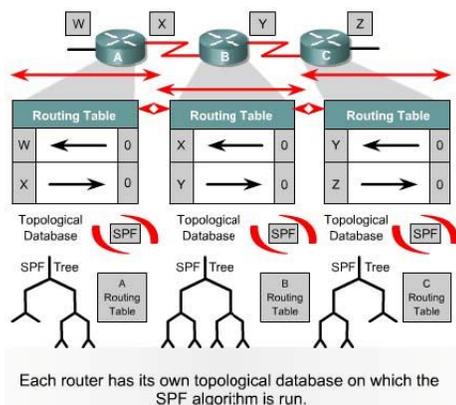
- b. *Topological database* – adalah kumpulan informasi yang dari *LSA-LSA*
- c. *SPF algorithm* – adalah hasil perhitungan pada database sebagai hasil dari pohon *SPF*
- d. *Routing table* – adalah daftar *route* dan *interface*



**Gambar 6. Konsep Link-State**

Ketika *router* melakukan pertukaran *LSA*, dimulai dengan jaringan yang terhubung langsung tentang informasi yang mereka miliki. Masing-masing *router* membangun *database topologi* yang berisi pertukaran informasi *LSA*.

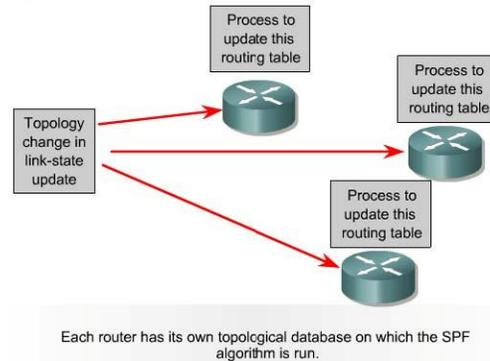
*Algoritma SPF* menghitung jaringan yang dapat dicapai. *Router* membangun *logical topologi* sebagai pohon (*tree*), dengan *router* sebagai *root*. *Topologi* ini berisi semua *route-route* yang mungkin untuk mencapai jaringan dalam *protokol link-state internetwork*. *Router* kemudian menggunakan *SPF* untuk memperpendek *route*. Daftar *route-route* terbaik dan *interface* ke jaringan yang dituju dalam *table routing*. *Link-state* juga memperbaiki *database topologi* yang lain dari elemen-elemen *topologi* dan status secara detail.



**Gambar 7. Jaringan Link-State Discovery**

*Router* pertama yang mempelajari perubahan *topologi link-state* melewati informasi sehingga

semua *router* dapat menggunakannya untuk proses *update*. adalah informasi *routing* dikirim ke semua *router* dalam *internetwork*. Untuk mencapai keadaan *konvergen*, setiap *router* mempelajari *router-router* tetangganya. Termasuk nama dari *router* tetangganya, status *interface* dan *cost* dari *link* ke tetangganya. *Router* membentuk paket *LSA* yang mendaftarkan informasi ini dari tetangga baru, perubahan *cost link* dan *link-link* yang tidak lagi *valid*. Paket *LSA* ini kemudian dikirim keluar sehingga semua *router* lain menerima itu.



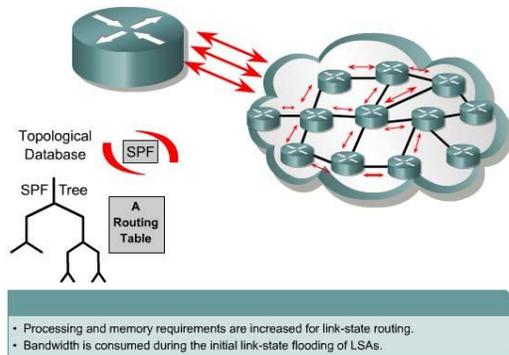
**Gambar 8. Perubahan Topologi Link-State**

Pada saat *router* menerima *LSA*, ia kemudian meng-*update table routing* dengan sebagian besar informasi yang terbaru. Data hasil perhitungan digunakan untuk membuat peta *internetwork* dan *algoritma SPF* digunakan untuk menghitung jalur terpendek ke jaringan lain. Setiap waktu paket *LSA* menyebabkan perubahan ke *database link-state*, kemudian *SPF* melakukan perhitungan ulang untuk jalur terbaik dan meng-*update table routing*. Ada beberapa titik berat yang berhubungan dengan *protokol link-state*:

- a. *Processor overhead*
- b. *Kebutuhan memori*
- c. *Konsumsi bandwidth*

*Router-router* yang menggunakan *protokol link-state* membutuhkan *memori* lebih dan proses data yang lebih dari pada *router-router* yang menggunakan *protokol distance vector*. *Router link-state* membutuhkan *memori* yang cukup untuk menangani semua informasi dari *database*, pohon *topologi* dan *table routing*. menunjukkan inisialisasi paket *flooding link-state* yang mengkonsumsi *bandwidth*. Pada proses inisial *discovery*, semua *router* yang menggunakan *protokol routing link-state* mengirimkan paket *LSA* ke semua *router* tetangganya. Peristiwa ini menyebabkan pengurangan *bandwidth* yang tersedia untuk *merouting trafik* yang membawa data *user*. Setelah inisial *flooding* ini, *protokol routing link-state* secara

umum membutuhkan *bandwidth* minimal untuk mengirim paket-paket *LSA* yang menyebabkan perubahan *topologi*.



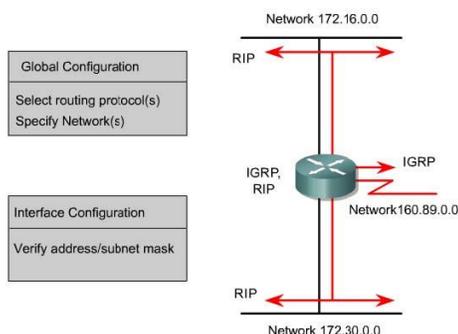
Gambar 9. Link-State Concern

Penentuan jalur *Router* menggunakan dua fungsi dasar: a. Fungsi penentuan jalur, b. Fungsi *switching*

Penentuan jalur terjadi pada *layer network*. Fungsi penentuan jalur menjadikan *router* untuk mengevaluasi jalur ke tujuan dan membentuk jalan untuk menangani paket. *Router* menggunakan *table routing* untuk menentukan jalur terbaik dan kemudian fungsi *switching* untuk melewatkan paket.

### 2.2. Konsep Link State

Dasar *algoritma routing* yang lain adalah *algoritma link state*. *Algoritma link state* juga bias disebut sebagai *algoritma Dijkstra* atau *algoritma shortest path first (SPF)*. Untuk menghidupkan *protokol routing* pada suatu *router*, membutuhkan *setting parameter global* dan *routing*. Tugas *global* meliputi pemilihan *protokol routing* seperti *RIP*, *IGRP*, *EIGRP* atau *OSPF*. Sedangkan tugas konfigurasi *routing* untuk menunjukkan jumlah jaringan *IP*. *Routing dinamis* menggunakan *broadcast* dan *multicast* untuk berkomunikasi dengan *router* lainnya.



Gambar 10. Tugas Konfigurasi Routing IP

### 2.3. OPNET Modeler 14.0

OPNET adalah salah satu *software* untuk *network modelling* yang sering digunakan dalam mendesain atau optimasi suatu jaringan. OPNET memiliki banyak modul yang disesuaikan dengan *equipment* dari banyak *vendor* yang digunakan pada banyak perusahaan. Dukungan inilah yang mempermudah *user* ataupun *designer* dalam merancang maupun melakukan optimasi suatu jaringan. Sayangnya fitur-fitur yang lengkap tersebut dijual per modul dengan harga yang cukup mahal

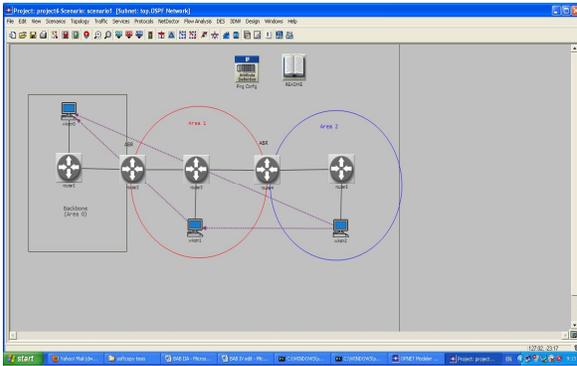


Gambar 11. Tampilan Menu OPNET Modeler 2.3.1. Simulasi Model

Di dalam simulasi jaringan berbasis *IP* dengan mempergunakan *simulator* OPNET Modeler 14.0, hal-hal yang perlu dilakukan antara lain:

#### 2.3.1.1. Konfigurasi Jaringan

*Software* OPNET Modeler 14.0 harus dilakukan penggambaran model jaringan yang akan disimulasikan. Konfigurasi yang digambarkan disesuaikan dengan model jaringan yang akan disimulasikan. Secara umum untuk menggambarkan suatu jaringan berbasis *IP* antara lain terdapat: *Router*, *bridge/switch*, *hub*, *LAN*, *link* baik yang dipergunakan untuk menghubungkan antar *router* ataupun hubungan ke *user*, *workstation*, *application server* dan komponen penunjang pada jaringan lainnya. Kelengkapan suatu model akan tergantung kepada kebutuhan dan kerumitan jaringan yang diinginkan.



Gambar 12. Model Jaringan

### 2.3.2. Tools OPNET

Merupakan fasilitas yang disediakan OPNET guna memudahkan pengguna dalam melakukan pekerjaan perancangan jaringan yang akan menyesuaikan dengan aslinya. Dalam *tools* OPNET fasilitas yang digunakan untuk perancangan awal biasa menggunakan antara lain:

#### 2.3.2.1. Network Model

Merupakan *scenario* dari bentuk *topologi* atau *infrastruktur* jaringan berupa *hardware* jaringan yang dihubungkan menggunakan media *transmisi*, yang di OPNET Modeler 14.0 dikenal dengan istilah *scenario network model*.

#### 2.3.2.2. Node Model

Merupakan cara kerja dari *node* yang diidentifikasi sehingga dalam *node* tersebut user dapat melakukan setingan dari yang seakan-akan merupakan jaringan sesungguhnya.

#### 2.3.2.3. Network Proses

Merupakan bentuk aplikasi yang scripnya ditulis menggunakan bahasa pemrograman C++ dimana user dapat mengetahui cara kerja dari router tersebut, Program ini dapat dimodifikasi sedemikian rupa jika peneliti ingin mengubah cara kerja dari *node* sistem peroutingan.

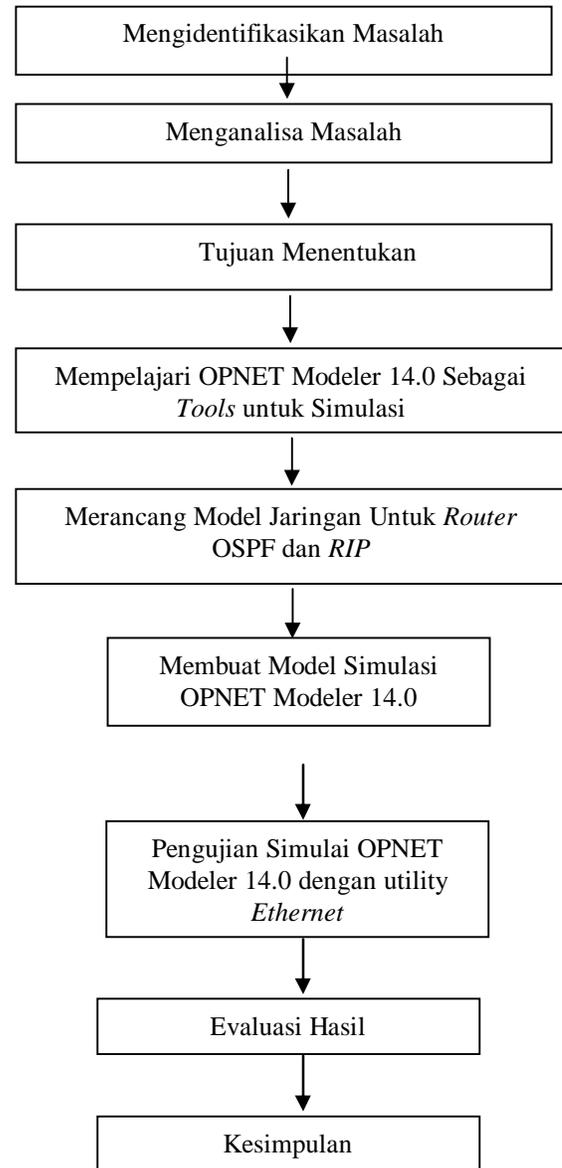
## 3. Metodologi Penelitian

Penelitian yang dilakukan merupakan satu bentuk kegiatan yang saling berkaitan dari satu kegiatan dengan kegiatan lainnya, maka tim merancang

kerangka kerja dalam penelitian ini secara *prosedure* sehingga setiap tahapan akan dapat dilihat pencapaian dan hasil akhir.

### 3.1. Kerangka Kerja

Kerangka kerja yang digunakan dalam penelitian ini adalah :



Gambar 13. Kerangka Kerja

### 3.2. Uraian Kerangka Kerja

Berdasarkan kerangka kerja, maka masing-masing langkahnya dapat diuraikan sebagai berikut :

### 3.2.1. Mendefenisikan Masalah

*Routing protocol* telah berkembang sesuai dengan perkembangan dunia teknologi jaringan komputer. Untuk menentukan yang lebih baik diantara *routing protocol* perlu adanya penelitian langsung, namun memerlukan biaya yang tidak sedikit jumlahnya, untuk mengantisipasi hal ini tim mencoba menganalisa *routing protocol* yang telah ada yaitu *OSPF* dan *RIP*.

### 3.2.2. Menganalisa Masalah

Masalah yang telah ada pada saat ini adalah perkembangan teknologi jaringan komputer sangat pesat dan berdampak pada ke tidak stabilisan jaringan dan seorang *administrator* mengalami kerumitan karena perubahan waktu pengiriman dan meneruskan data mengalami dinamika. Dengan mengidentifikasi *rute-rute* pada pengalaman dari satu *router* ke *router* lainnya dapat membantu mempercepat dan meneruskan data.

### 3.2.3. Menentukan Tujuan

Tujuan akhir yang ingin dicapai oleh tim adalah memberikan informasi kepada *administrator* sebagai pertimbangan dalam mendesign jaringan yang memiliki kemudahan dalam pengoperasiannya seperti *routing protocol OSPF* yang bekerja secara dinamis yang akan menentukan *rute* terdekat untuk menghubungkan satu *router* dengan *router* lainnya.

### 3.2.4. Mempelajari OPNET Modeler 14.0 Sebagai Tools untuk Simulasi

Untuk mencapai tujuan, maka dipelajari *Software OPNET Modeler 14.0* sebagai *Tools* yang mana digunakan sebagai media simulasi dalam penelitian. Untuk menggunakan *OPNET Modeler 14.0* harus memiliki pengetahuan akan *tools* yang berada didalamnya dan dapat menentukan *utility* yang dapat digunakan sebagai parameter pengukuran terhadap *routing protocol* dalam penelitian ini tim menganalisa dan membandingkan *Load (bits/sec)* antara *routing protocol OSPF* dan *RIP*.

### 3.2.5. Merancang Model untuk Router OSPF dan RIP

*Routing protocol OSPF* dan *RIP* merupakan *protocol antonomous* sistem dan menggunakan *link state* sehingga keduanya akan sulit di perkirakan yang *protokol* mana lebih baik, tim mencoba merancang satu *topologi* jaringan yang dapat dipergunakan untuk membandingkan diantara kedua *routing protocol* tersebut. Dengan membedakan jalur sehingga didapatkan perubahan waktu meneruskan data.

### 3.2.6. Membuat Model Simulasi OPNET Modeler 14.0

Dari rancangan yang telah dibuat tim menggunakan rancangan *network* model simulasi *OPNET Modeler 14.0*. untuk dapat menyesuaikan dengan kondisi seakan-akan sesungguhnya dan dapat digunakan untuk menganalisa *routing protocol* tersebut tim mengubah *network* model tersebut.

### 3.2.7. Pengujian

Bagian ini melakukan pengujian model dengan *software* Simulasi *OPNET Modeler 14.0*. Dalam pengujian tim menggunakan *Load (bits/sec)* yang telah disediakan pada *software* tersebut dengan menggunakan model jaringan dan *utility* yang sama yaitu *ethernet*, Yang dapat ditampilkan dengan perbandingan angka dan grafik.

### 3.2.8. Evaluasi Hasil

Bagian ini bertujuan melihat dan mengumpulkan hasil dari penelitian dengan membandingkan *routing protocol OSPF* dan *RIP* yang dapat dilihat dalam format grafik dari masing *routing protocol* berdasarkan *utility ethernet*.

Dari evaluasi yang di dapat dalam simulasi tim mengharap dapat menampilkan kelebihan dan kelemahan dari masing-masing *routing protocol* sehingga dapat membantu *administrator* dalam membangun jaringan komputer yang kompleks.

## 4. Analisa Dan Implementasi Sistem

Dalam jaringan komputer banyak komponen dapat menyebabkan perubahan waktu penyampain data baik *hardware* komputer, *software* komputer maupun *hardware* pendukung jaringan komputer.

Menjawab penyebab perubahan waktu penyampaian dapat terjadi *hardware* pada jaringan komputer dengan alasan :

1. *Latency* merupakan *delay* yang terjadi pada satu *node* dimana proses terjadi sehingga juga dapat memperlambat proses.
2. *Delay host to host* merupakan suatu bentuk terjadinya jeda waktu data yang dikirim dari sumber ke tujuan.
3. *Load* merupakan beban yang melalui jaringan tidak selalu konstan dimana akan berubah-ubah sehingga mempengaruhi waktu data tiba ke tujuan. Untuk memberi beban pada client menggunakan distribusi probabilitas normal yang memberi sebaran data dalam *range* waktu data sampai ketujuan.
4. *Bandwidth* yang di gunakan pada jaringan berpengaruh pada waktu data yang akan diterima *server*, menyesuaikan dengan besar atau kecil data sehingga data yang akan dilewatkan sampai ke tujuan

Berdasarkan analisa diatas, dasar tim lebih menspesifikkan yang akan di teliti adalah membandingkan kemampuan melewatkan data pada *Routing Protocol* dengan menggunakan *variabel load (bits/sec)*

#### 4.1. Mode Traffic Data

Data yang digunakan adalah merupakan di *design* menyerupai kondisi jaringan sesungguhnya dengan melewatkan data yang bervariasi pada proses simulasi, diantaranya :

1. *Operation mode traffic* menggunakan *Serial Random* yang telah ada pada. *Serial random* digunakan dengan asumsi *traffic* yang lewat pada jaringan komputer tidaklah dapat diprediksi besarnya (*bits*)
2. *Duration* yang diteliti merupakan waktu melewatkan data yang di *load (bits)* dari *client* dan di *received* oleh *server* yang merupakan ukuran data lengkap sama dengan *client*.
3. *Distribusi probabilitas statistik* pada *software* simulasi menggunakan distribusi Normal, dimana untuk mengetahui maximal dan minimal data yang terkirim dalam selama proses melewatkan data.

Sistem *peroutingan protocol* dalam jaringan komputer juga akan mengalami perubahan waktu menyesuaikan *load* (beban) yang melalui jaringan.

sehingga *user* sering kali menemukan perbedaan waktu saat *client* mengirim data dalam satu jaringan yang sama. Tim mencoba merancang, mengembangkan dan menganalisa khususnya cara kerja *routing protocol*. *Routing protocol* yang digunakan sebagai objek adalah :

1. *OSPF ( Open Shortest Path First )* merupakan inovasi Sistem peroutingan yang dapat mengidentifikasi jalur terpendek atau terdekat dalam melewatkan data dari sumber ke tujuannya, hal ini memungkinkan kinerja jaringan akan lebih ringan dan proses dapat lebih cepat.
2. *RIP ( Routing Internet Protocol )* merupakan *protocol* yang paling banyak digunakan *router* pada product *Cisco*, dan product ini telah lebih dahulu mendapat kepercayaan masyarakat pada sistem *routing* di jaringan berbasis *LAN*. Walau demikian sistem *routing* yang digunakan dalam melewatkan paket data hanya dapat mengidentifikasi jaringan yang terdekat dan untuk selanjutnya akan dilakukan hal yang sama sampai data ketujuan atau hanya memilih *node* yang paling sedikit dalam satu jalur.

#### 4.2. Implementasi Sistem

Untuk melihat kemampuan melewatkan data pada *Scenario Network Model* dengan menggunakan 3 jalur dan *input* yang sama maka tim akan bisa membuktikan analisa dan perancangan yang telah dibahas pada bab IV. Dalam simulasi ini tim menggunakan OPNET Modeler 14.0 karena merupakan salah satu simulasi yang di pergunakan untuk melihat *output* pada jaringan komputer.

#### 4.3. Hasil Implementasi

Untuk mendapatkan hasil penelitian tim membagi dalam 3 jalur *peroutingan* . setiap analisa dibagi kembali berdasarkan *node* dalam setiap jalurnya yaitu:

1. Jalur 1
  - a. *Client*
  - b. *Router 1, Router 9, Router 2,*
  - c. *Sever*
2. Jalur 2
  - a. *Client*
  - b. *Router 1, Router 3, Router 4, dan Router 2*
  - c. *Router Server*
3. Jalur 3
  - a. *Client*

- b. Router 1, Router 5, Router 6, Router 7, Router 8, dan Router 2
- c. Server

**4.4. Results Browser**

Merupakan tampilan dari hasil *input* atau *output* seluruh *node* yang telah di rancang dan di *setting* berdasarkan fungsi dan kebutuhan dalam penelitian ini. Tampilan grafik yang digunakan ada 2 yaitu :

1. *Stacket Statistics* adalah menampilkan *node* pada satu grafik
2. *Overlaid Statistics* adalah menampilkan beberapa *node* dalam satu grafik

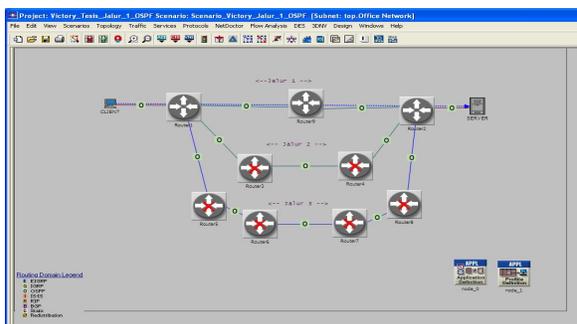
**4.5. Scenario Network Model Protocol OSPF Jalur 1**

*Scenario Network Model* Jalur 1 di gunakan untuk melewati data (*load*) secara keseluruhan (data *drop* pada *server*), *Node* pada jalur 1 terdiri atas :

- a. *Client*
- b. Router 1, Router 9 dan Router 2
- c. *Server*

*Scenario Network Model* Jalur 1 di pisahkan agar dalam simulasi analisa output hanya menampilkan jalur 1saja, jalur yang di pisahkan antara lain :

- a. Menonaktifkan jalur 2
- b. Dan menonaktifkan jalur 3



**Gambar 14. Scenario Network Model Protocol OSPF Jalur 1**

**4.5.1. Results Browser Routing Protocol OSPF Jalur 1**

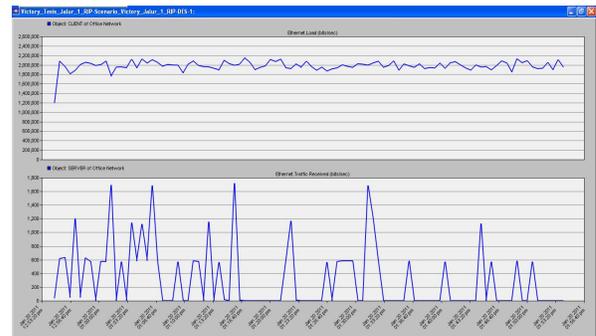
Merupakan tampilan dari hasil *input* atau *output* yang telah di rancang dan di *setting* berdasarkan fungsi dan kebutuhan *Results Browser* Jalur 1.

**Tabel 1. Results Browser Routing Protocol OSPF Jalur 1**

Jalur	Protocol	Stacket Statistics		Overlaid Statistics
		Client	Server	Client & Server
1	Load	√	√	√
1	Recieved	√	√	

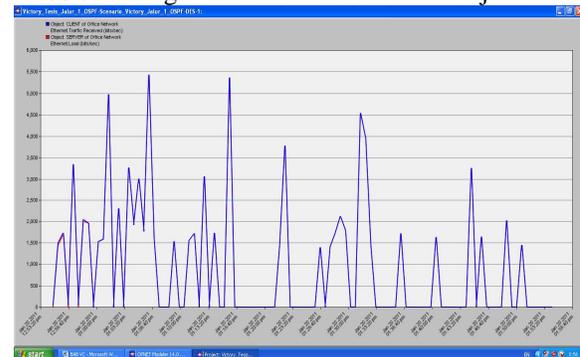
**4.5.2. Overlaid Statistics Data Client dan Server Protocol OSPF Jalur 1 Duration 1 Jam**

Membandingkan tampilan data (*load*) yang bersumber dari *client* dan *server* menerima data (*traffic received (bit/sec)*) tersebut yang sebelumnya melalui Router 1, Router 9 dan Router 2 data yang diterima sekitar 50% yaitu dari *maximal* data 2.170.000 bits dan *minimal* data 1.800.000 bits sedangkan data yang diterima *server* dari *maximal* 5.000 *bits* dan *minimal* 1.400 *bits*.



**Gambar 15. Overlaid Statistics Data Client dan Server Protocol OSPF Jalur 1 Duration 1 Jam**

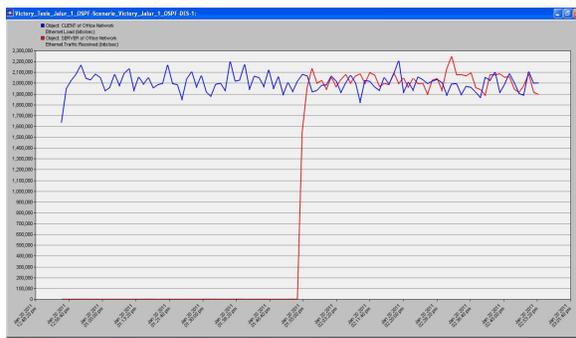
Namun data yang dilewatkan dari *client* (*received*) sampai ke *server* (*load*) dengan kapasitas data yang sama dari *maximal* 5.400 *bits* dan *minimal* 1.400 *bits* dengan waktu simulasi selama 1 jam.



**Gambar 16. Overlaid Statistics Data Client (received) dan Server (load) Protocol OSPF Jalur 1 Duration 1 Jam**

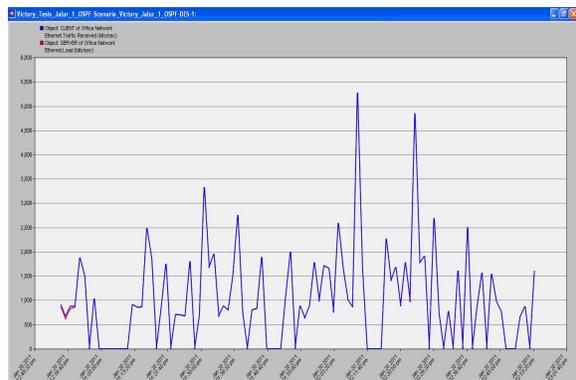
#### 4.5.3. Overlaid Statistics Data Client dan Server Protocol OSPF Jalur 1 Duration 2 Jam

Membandingkan tampilan data (load) yang bersumber dari *client* dan *server* menerima data ( *traffic received (bit/sec)* ) tersebut yang sebelumnya melalui *Router 1*, *Router 9* dan *Router 2* data yang diterima sekitar 75% yaitu dari *maximal* data 2.200.000 *bits* dan *minimal* data 1.825.000 *bits* sedangkan data yang diterima *server* dari *maximal* 2.250.000 *bits* dan *minimal* 1.900.000 *bits*.



**Gambar 17. Overlaid Statistics Data Client dan Server Protocol OSPF Jalur 1 Duration 2 Jam**

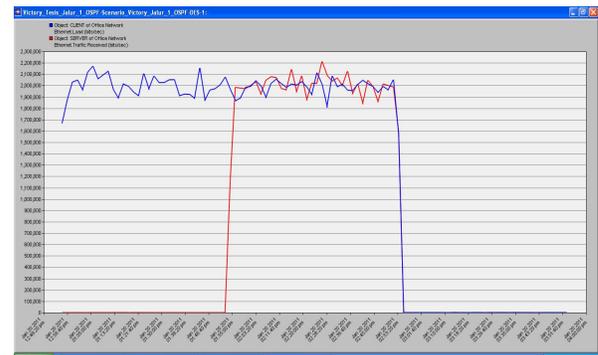
Namun data yang dilewatkan dari *client* (received) sampai ke *server* (load) dengan kapasitas data yang sama dari *maximal* 5.300 *bits* dan *minimal* 700 *bits* dengan waktu simulasi selama 1 jam.



**Gambar 18. Overlaid Statistics Data Client (received) dan Server (load) Protocol OSPF Jalur 1 Duration 2 Jam**

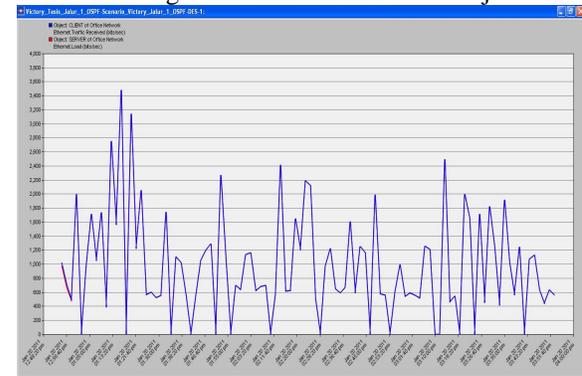
#### 4.5.4. Overlaid Statistics Data Client dan Server Protocol OSPF Jalur 1 Duration 3 Jam

Membandingkan tampilan data (load) yang bersumber dari *client* dan *server* menerima data ( *traffic received (bit/sec)* ) tersebut yang sebelumnya melalui *Router 1*, *Router 9* dan *Router 2* data yang diterima sekitar 100% yaitu dari *maximal* data 2.170.000 *bits* dan *minimal* data 1.800.000 *bits* sedangkan data yang diterima *server* dari *maximal* 5.000 *bits* dan *minimal* 1.400 *bits*.



**Gambar 19. Overlaid Statistics Data Client dan Server Protocol OSPF Jalur 1 Duration 3 Jam**

Namun data yang dilewatkan dari *client* (received) sampai ke *server* (load) dengan kapasitas data yang sama dari *maximal* 5.000 *bits* dan *minimal* 1.400 *bits* dengan waktu simulasi selama 3 jam.



**Gambar 20. Overlaid Statistics Data Client (received) dan Server (load) Protocol OSPF Jalur 1 Duration 3 Jam**

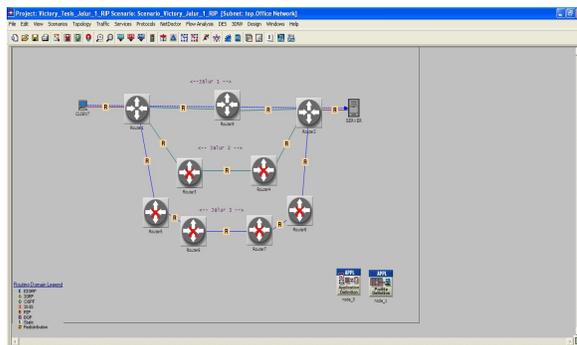
#### 4.6. Scenario Network Model Protocol RIP Jalur 1

Scenario Network Model Jalur 1 di gunakan untuk melewati data (load) secara keseluruhan (data drop pada server), Node pada jalur 1 terdiri atas :

- a. Client
- b. Router 1, Router 9, Router 2
- c. Server

Scenario Network Model Jalur 1 di pisahkan agar dalam simulasi analisa output hanya menampilkan jalur 1saja, jalur yang di pisahkan antara lain :

- a. Menonaktifkan jalur 2
- b. Dan menonaktifkan jalur 3



Gambar 21. Scenario Network Model Protocol RIP Jalur 1

4.6.1. Results Browser Routing Protoco RIP Jalur 1

Pada jalur 1 ini menampilkan grafik load dan received pada node server, yang mana received data yang pertama kali tiba di server sedangkan load mengidentifikasi data ditampilkan.

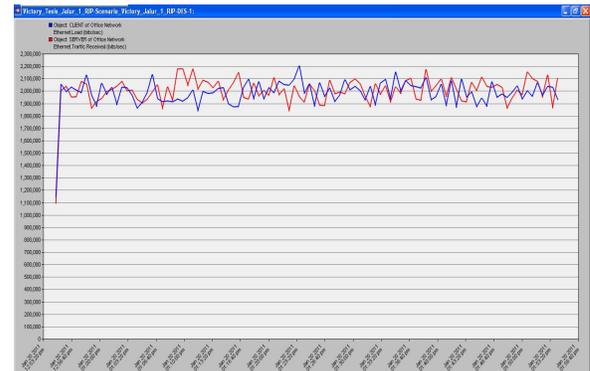
Tabel 2. Results Browser Routing Protocol RIP Jalur 1

Jalur	Protocol	Stacket Statistics		Overlaid Statistics
		Client	Server	Client & Server
2	Load	√	√	√
2	Recieved	√	√	

4.6.2. Overlaid Statistics Data Client dan Server Protocol RIP Jalur 1 Duration 1 Jam

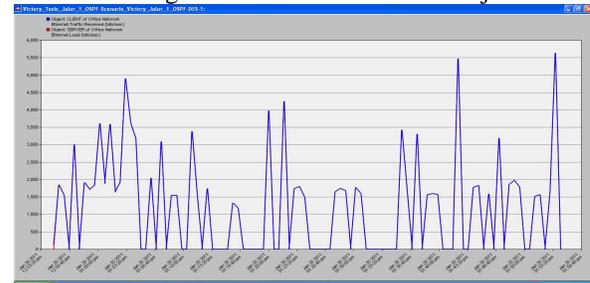
Membandingkan tampilan data (load) yang bersumber dari client dan server menerima data (

traffic received (bit/sec) ) tersebut yang sebelumnya melalui Router 1, Router 9 dan Router 2 data yang diterima sekitar 50% yaitu dari maximal data 2.200.000 bits dan minimal data 1.850.000 bits sedangkan data yang diterima server dari maximal 2.190.000 bits dan minimal 1.850.000 bits.



Gambar 22. Overlaid Statistics Data Client dan Server Protocol RIP Jalur 1 Duration 1 Jam

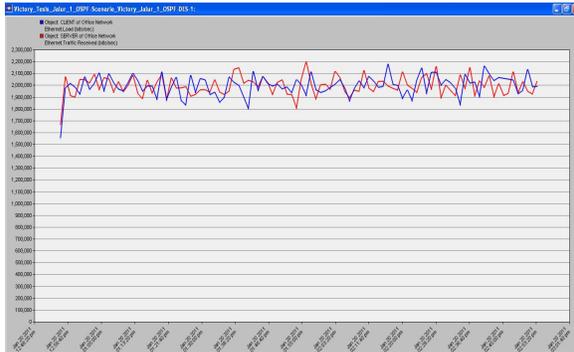
Namun data yang dilewatkan dari client (received) sampai ke server (load) dengan kapasitas data yang sama dari maximal 5.550 bits dan minimal 1.800 bits dengan waktu simulasi selama 1 jam.



Gambar 23. Overlaid Statistics Data Client (received) dan Server (load) Protocol RIP Jalur 1 Duration 1 Jam

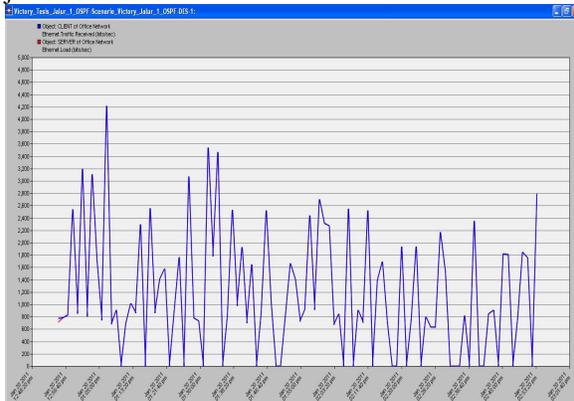
4.6.3. Overlaid Statistics Data Client dan Server Protocol RIP Jalur 1 Duration 2 Jam

Membandingkan tampilan data (load) yang bersumber dari client dan server menerima data ( traffic received (bit/sec) ) tersebut yang sebelumnya melalui Router 1, Router 9 dan Router 2 data yang diterima sekitar 75% yaitu dari maximal data 2.190.000 bits dan minimal data 1.800.000 bits sedangkan data yang diterima server dari maximal 2.200.000 bits dan minimal 1.800.000 bits.



**Gambar 24. Overlaid Statistics Data Client dan Server Protocol RIP Jalur 1 Duration 2 Jam**

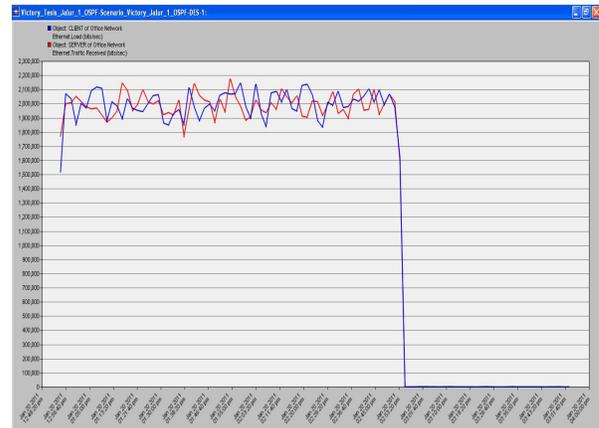
Namun data yang dilewatkan dari *client* (*received*) sampai ke *server* (*load*) dengan kapasitas data yang sama dari maximal 4.200 *bits* dan minimal 800 *bits* dengan waktu simulasi selama 1 jam.



**Gambar 25. Overlaid Statistics Data Client (received) dan Server Protocol RIP (load) Jalur 1 Duration 2 Jam**

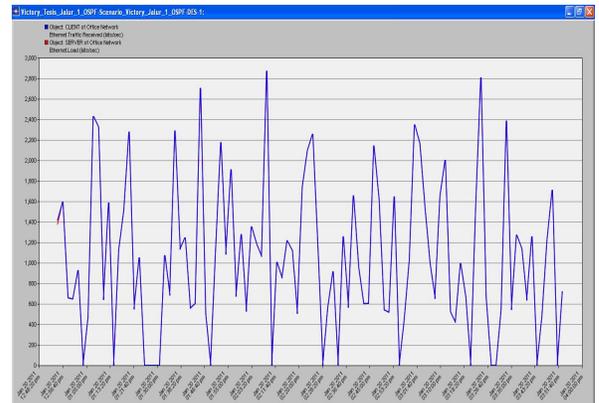
#### 4.6.4. Overlaid Statistics Data Client dan Server Protocol RIP Jalur 1 Duration 3 Jam

Membandingkan tampilan data (*load*) yang bersumber dari *client* dan *server* menerima data (*traffic received (bit/sec)*) tersebut yang sebelumnya melalui *Router 1*, *Router 9* dan *Router 2* data yang diterima sekitar 100% yaitu dari maximal data 2.150.000 *bits* dan minimal data 1.790.000 *bits* sedangkan data yang diterima *server* dari maximal 2.190.000 *bits* dan minimal 1.790.000 *bits*.



**Gambar 26. Overlaid Statistics Data Client dan Server Protocol RIP Jalur 1 Duration 3 Jam**

Namun data yang dilewatkan dari *client* (*received*) sampai ke *server* (*load*) dengan kapasitas data yang sama dari maximal 2.850 *bits* dan minimal 900 *bits* dengan waktu simulasi selama 3 jam.



**Gambar 27. Overlaid Statistics Data Client (received) dan Server (load) Protocol RIP Jalur 1 Duration 3 Jam**

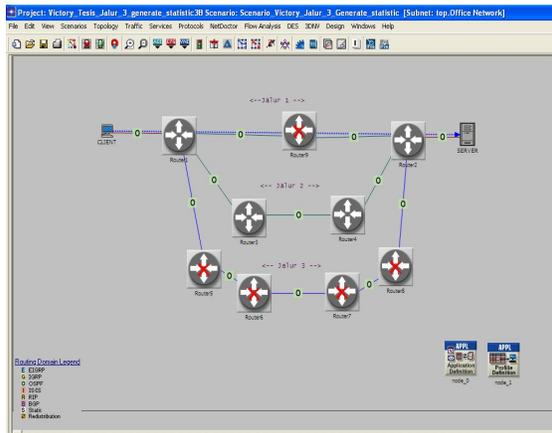
#### 4.7. Scenario Network Model Protocol OSPF Jalur 2

*Scenario Network Model Jalur 2* di gunakan untuk melewatkan data (*load*) secara keseluruhan (data *drop* pada *server*), Node pada jalur 2 terdiri atas :

- Client*
- Router 1*, *Router 3*, *Router 4*, *Router 2*
- Server*

Scenario Network Model Jalur 2 di pisahkan agar dalam simulasi analisa output hanya menampilkan jalur 2 saja, jalur yang di pisahkan antara lain :

- a. Menonaktifkan jalur 1
- b. Dan menonaktifkan jalur 3



Gambar 28. Scenario Network Model Protocol OSPF Jalur 2

#### 4.7.1. Results Browser Routing Protocol OSPF Jalur 2

Merupakan tampilan dari hasil *input* atau *output server* yang telah di rancang dan di setting berdasarkan fungsi dan kebutuhan *Results Browser Jalur 2*.

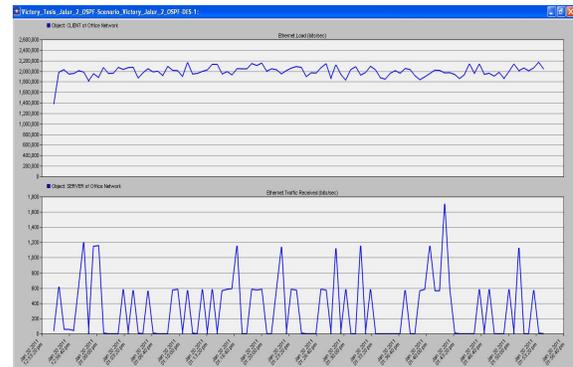
Tabel 3. Results Browser Routing Protocol OSPF Jalur 2

Jalur	Protocol	Stacket Statistics		Overlaid Statistics
		Client	Server	Client & Server
2	Load	√	√	
2	Received	√	√	√

#### 4.7.2. Overlaid Statistics Data Client dan Server Protocol OSPF Jalur 2 Duration 1 Jam

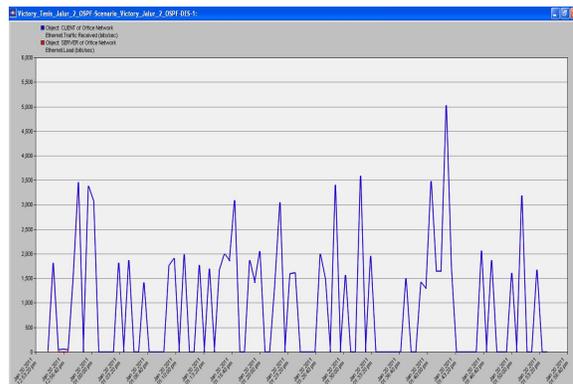
Membandingkan tampilan data (*load*) yang bersumber dari *client* dan *server* menerima data (*traffic received (bit/sec)*) tersebut yang sebelumnya melalui *Router 1, Router 3, Router 4* dan *Router 2* data yang diterima sekitar 50% yaitu dari *maximal* data 2.200.000 *bits* dan *minimal* data 1.800.000 *bits*

sedangkan data yang diterima *server* dari maximal 1.700 *bits* dan minimal 590 *bits*.



Gambar 29. Overlaid Statistics Data Client dan Server Protocol OSPF Jalur 2 Duration 1 Jam

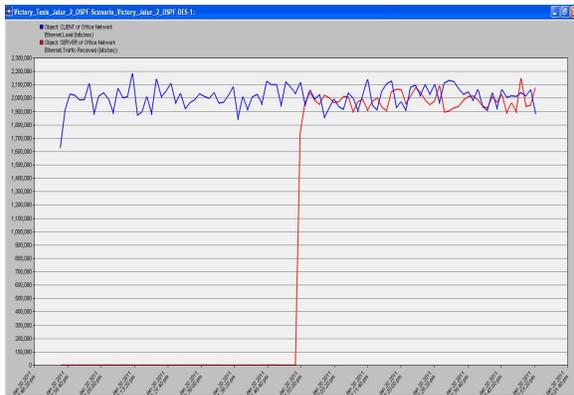
Namun data yang dilewatkan dari *client* (*received*) sampai ke *server* (*load*) dengan kapasitas data yang sama dari *maximal* 5.000 *bits* dan *minimal* 1.400 *bits* dengan waktu simulasi selama 1 jam.



Gambar 30. Overlaid Statistics Data Client (*received*) dan Server (*load*) Protocol OSPF Jalur 2 Duration 1 Jam

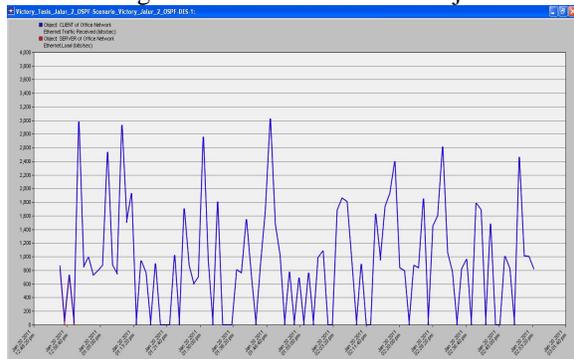
#### 4.7.3. Overlaid Statistics Data Client dan Server Protocol OSPF Jalur 2 Duration 2 Jam

Membandingkan tampilan data (*load*) yang bersumber dari *client* dan *server* menerima data (*traffic received (bit/sec)*) tersebut yang sebelumnya melalui *Router 1, Router 3, Router 4* dan *Router 2* data yang diterima sekitar 75% yaitu dari *maximal* data 2.190.000 *bits* dan *minimal* data 1.850.000 *bits* sedangkan data yang diterima *server* dari maximal 2.050.000 *bits* dan minimal 1.900.000 *bits*.



**Gambar 31. Overlaid Statistics Data Client dan Server Protocol OSPF Jalur 2 Duration 2 Jam**

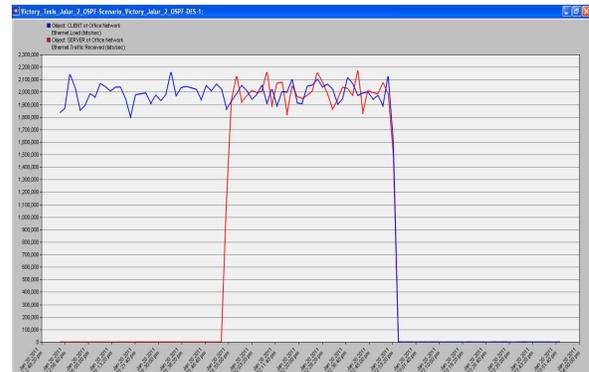
Namun data yang dilewatkan dari *client* (*received*) sampai ke *server* (*load*) dengan kapasitas data yang sama dari *maximal* 3.000 *bits* dan minimal 650 *bits* dengan waktu simulasi selama 2 jam..



**Gambar 32. Overlaid Statistics Data Client (received) dan Server (load) Protocol OSPF Jalur 2 Duration 2 Jam**

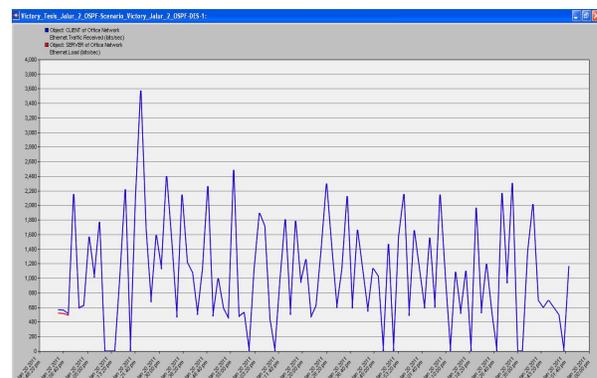
#### 4.7.4. Overlaid Statistics Data Client dan Server Protocol OSPF Jalur 2 Duration 3 Jam

Membandingkan tampilan data (*load*) yang bersumber dari *client* dan *server* menerima data (*traffic received (bit/sec)*) tersebut yang sebelumnya melalui *Router 1*, *Router 3*, *Router 4* dan *Router 2* data yang diterima sekitar 100% yaitu dari *maximal* data 2.180.000 *bits* dan *minimal* data 1.800.000 *bits* sedangkan data yang diterima *server* dari *maximal* 2.180.000 *bits* dan *minimal* 1.810.000 *bits*.



**Gambar 33. Overlaid Statistics Data Client dan Server Protocol OSPF Jalur 2 Duration 3 Jam**

Namun data yang dilewatkan dari *client* (*received*) sampai ke *server* (*load*) dengan kapasitas data yang sama dari *maximal* 3.590 *bits* dan minimal 1.000 *bits* dengan waktu simulasi selama 2 jam.



**Gambar 34. Overlaid Statistics Data Client (received) dan Server (load) Protocol OSPF Jalur 2 Duration 3 Jam**

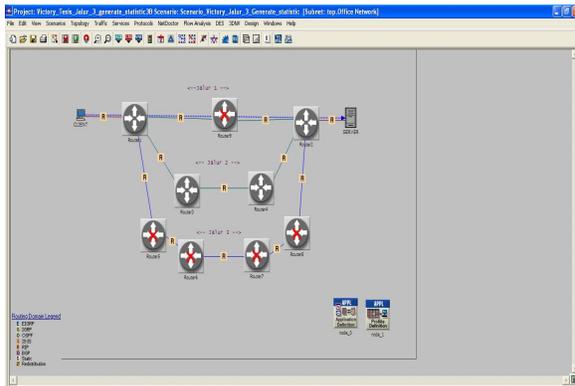
#### 5.7.Scenario Network Model Protocol RIP Jalur 2

*Scenario Network Model* Jalur 2 di gunakan untuk melewatkan data (*load*) secara keseluruhan (*data drop* pada *server*), Node pada jalur 2 terdiri atas :

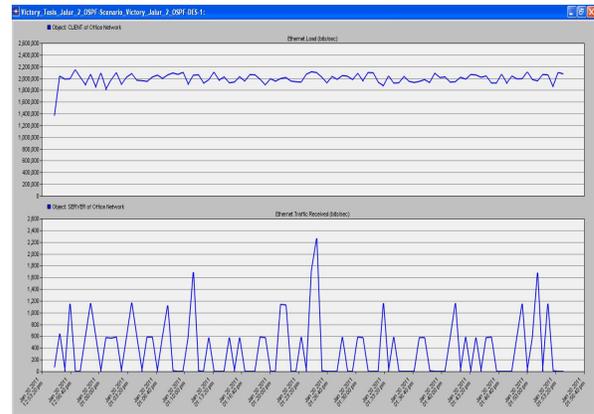
- Client*
- Router 1*, *Router 3*, *Router 4* dan *Router 2*
- Server*

*Scenario Network Model* Jalur 2 di pisahkan agar dalam simulasi analisa output hanya menampilkan jalur 2saja, jalur yang di pisahkan antara lain :

- Menonaktifkan jalur 1
- Dan menonaktifkan jalur 3



Gambar 35. Scenario Network Model Protocol RIP Jalur 2



Gambar 36. Overlaid Statistics data Client dan Server Protocol RIP Jalur 2 Duration 1 Jam

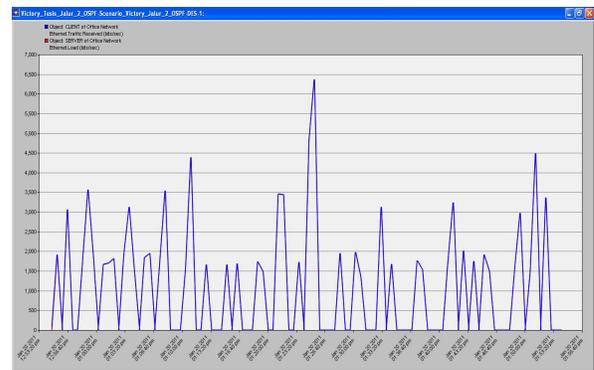
Namun data yang dilewatkan dari *client* (*received*) sampai ke *server* (*load*) dengan kapasitas data yang sama dari *maximal* 6.400 *bits* dan *minimal* 1.700 *bits* dengan waktu simulasi selama 1 jam.

5.8.1. Results Browser Routing Protocol RIP Jalur 2

Merupakan tampilan hasil *input* atau *output* seluruh *node* yang telah di rancang dan di setting berdasarkan fungsi dan kebutuhan *Results Browser* Jalur 2.

Tabel 4. Results Browser Routing Protocol RIP Jalur 2

Jalur	Protocol	Stacket Statistics		Overlaid Statistics
		Client	Server	Client & Server
2	Load	√	√	√
2	Recieved	√	√	



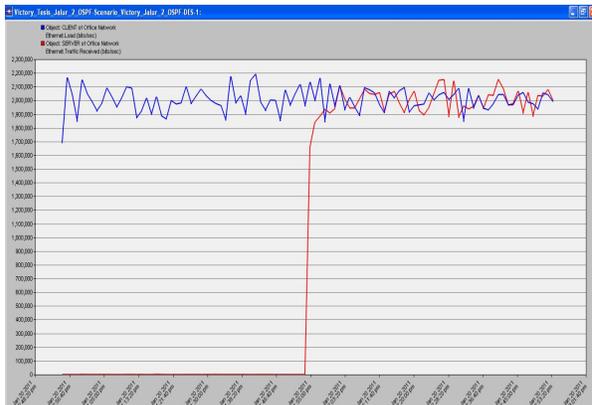
Gambar 37. Overlaid Statistics Data Client (received) dan Server (load) Protocol RIP Jalur 2 Duration 1 Jam

5.8.2. Overlaid Statistics Data Client dan Server Protocol RIP Jalur 2 Duration 1 Jam

Membandingkan tampilan data (*load*) yang bersumber dari *client* dan *server* menerima data (*traffic received (bit/sec)*) tersebut yang sebelumnya melalui *Router 1*, *Router 3*, *Router 4* dan *Router 2* data yang diterima sekitar 50% yaitu dari *maximal* data 2.190.000 *bits* dan *minimal* data 1.800.000 *bits* sedangkan data yang diterima *server* dari *maximal* 2.300 *bits* dan *minimal* 600 *bits*.

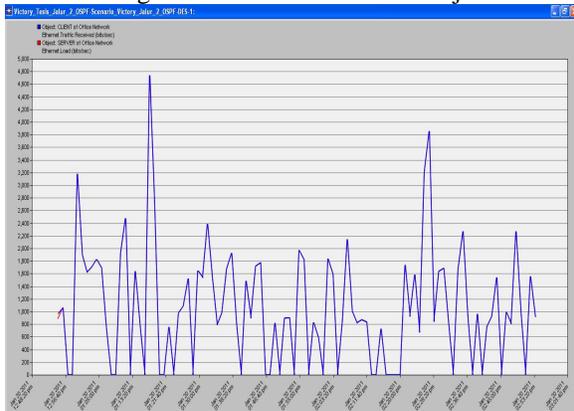
5.8.3. Overlaid Statistics Data Client dan Server Protocol RIP Jalur 2 Duration 2 Jam

Membandingkan tampilan data (*load*) yang bersumber dari *client* dan *server* menerima data (*traffic received (bit/sec)*) tersebut yang sebelumnya melalui *Router 1*, *Router 3*, *Router 4* dan *Router 2* data yang diterima sekitar 75% yaitu dari *maximal* data 2.200.000 *bits* dan *minimal* data 1.880.000 *bits* sedangkan data yang diterima *server* dari *maximal* 2.150.000 *bits* dan *minimal* 1.890.000 *bits*



**Gambar 38. Overlaid Statistics Data Client dan Server Protocol RIP Jalur 2 Duration 2 Jam**

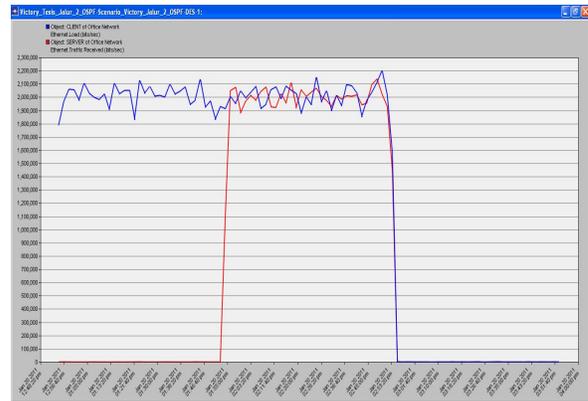
Namun data yang dilewatkan dari *client* (*received*) sampai ke *server* (*load*) dengan kapasitas data yang sama dari *maximal* 4.700 bits dan *minimal* 700 bits dengan waktu simulasi selama 2 jam.



**Gambar 39. Overlaid Statistics Data Client (received) dan Server (load) Protocol RIP Jalur 2 Duration 2 Jam**

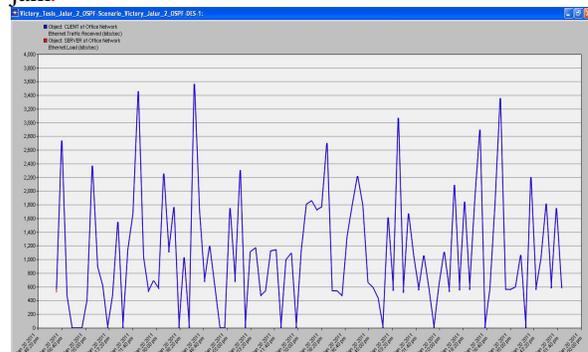
#### 5.8.4. Overlaid Statistics Data Client dan Server Protocol RIP Jalur 2 Duration 3 Jam

Membandingkan tampilan data (*load*) yang bersumber dari *client* dan *server* menerima data (*traffic received (bit/sec)*) tersebut yang sebelumnya melalui *Router 1*, *Router 3*, *Router 4* dan *Router 2* data yang diterima sekitar 100% yaitu dari *maximal* data 2.220.000 bits dan *minimal* data 1.850.000 bits sedangkan data yang diterima *server* dari *maximal* 2.150.000 bits dan *minimal* 1.900.000 bits.



**Gambar 40. Overlaid Statistics Data Client dan Server Protocol RIP Jalur 2 Duration 3 Jam**

Namun data yang dilewatkan dari *client* (*received*) sampai ke *server* (*load*) dengan kapasitas data yang sama dari *maximal* 3.590 bits dan *minimal* 1.100 bits dengan waktu simulasi selama 3 jam.



**Gambar 41. Overlaid Statistics Data Client (received) dan Server (load) Protocol RIP Jalur 2 Duration 3 Jam**

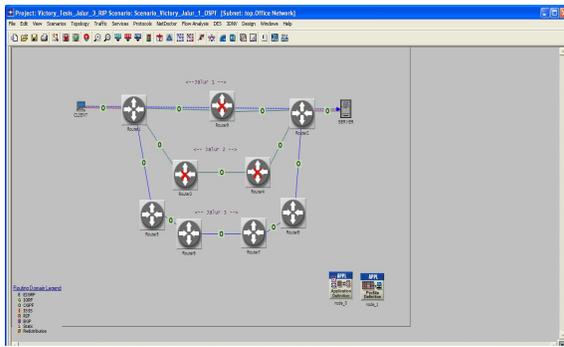
### 5.8. Scenario Network Model OSPF Jalur 3

*Scenario Network Model Jalur 3* di gunakan untuk melewatkan data (*load*) secara keseluruhan (*data drop* pada *server*), Node pada jalur 3 terdiri atas :

- Client*
- Router 1*, *Router 5*, *Router 6*, *Router 7*, *Router 8*, dan *Router 2*
- Server*

*Scenario Network Model Jalur 3* di pisahkan agar dalam simulasi analisa output hanya menampilkan jalur 3 saja, jalur yang di pisahkan antara lain :

- Menonaktifkan jalur 1
- Dan menonaktifkan jalur 2



Gambar 42. Scenario Network Model Protocol OSPF Jalur 3

### 5.9.1. Results Browser Routing Protocol OSPF Jalur 3

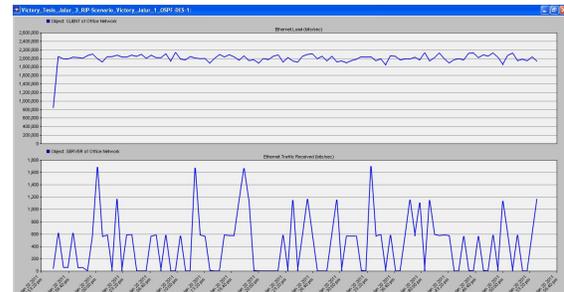
Merupakan tampilan hasil *input* atau *output* seluruh *node* yang telah di rancang dan di setting berdasarkan fungsi dan kebutuhan *Results Browser* Jalur 3.

Tabel 5. Results Browser Routing Protocol OSPF Jalur 3

Jalur	Protocol	Stacket Statistics		Overlaid Statistics
		Client	Server	Client & Server
3	Load	√	√	√
3	Recieved	√	√	

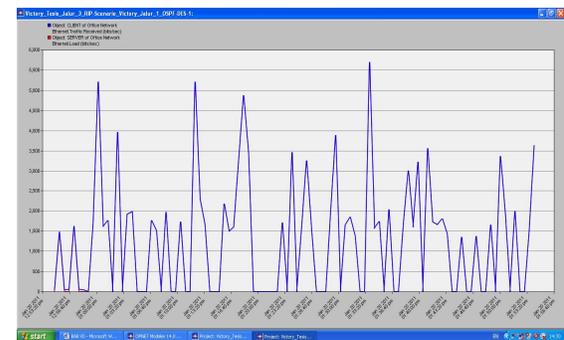
### 5.9.2. Overlaid Statistics Data Client dan Server Protocol OSPF Jalur 3 Duration 1 Jam

Membandingkan tampilan data (*load*) yang bersumber dari client dan server menerima data ( *traffic received (bit/sec)* ) tersebut yang sebelumnya melalui *Router 1, Router 5, Router 6, Router 7, Router 8 dan Router 2* data yang diterima sekitar 75% yaitu dari *maximal* data 2.200.000 *bits* dan *minimal* data 1.850.000 *bits* sedangkan data yang diterima *server* dari *maximal* 5.550 *bits* dan *minimal* 1.300 *bits*



Gambar 43. Overlaid Statistics Data Client dan Server Protocol OSPF Jalur 3 Duration 1 Jam

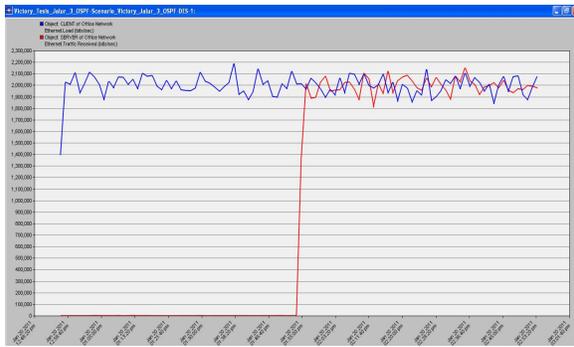
Namun data yang dilewatkan dari *client* (*received*) sampai ke *server* (*load*) dengan kapasitas data yang sama dari *maximal* 5.550 *bits* dan *minimal* 1.400 *bits* dengan waktu simulasi selama 1 jam



Gambar 44. Overlaid Statistics Data Client (*received*) dan Server (*load*) Protocol OSPF Jalur 3 Duration 1 Jam

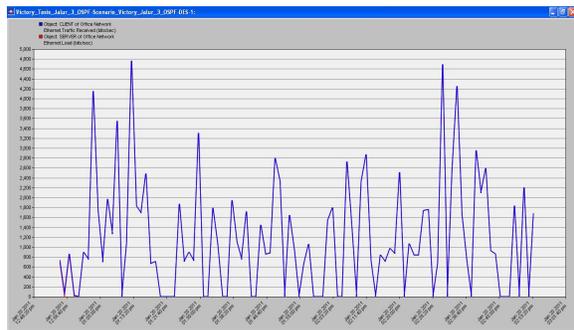
### 5.9.3. Overlaid Statistics Data Client dan Server Protocol OSPF Jalur 3 Duration 2 Jam

Membandingkan tampilan data (*load*) yang bersumber dari *client* dan *server* menerima data ( *traffic received (bit/sec)* ) tersebut yang sebelumnya melalui *Router 1, Router 5, Router 6, Router 7, Router 8 dan Router 2* data yang diterima sekitar 75% yaitu dari *maximal* data 2.200.000 *bits* dan *minimal* data 1.850.000 *bits* sedangkan data yang diterima *server* dari *maximal* 2.150.000 *bits* dan *minimal* 1.800.000 *bits*



**Gambar 45. Overlaid Statistics Data Client dan Server Protocol OSPF Jalur 3 Duration 2 Jam**

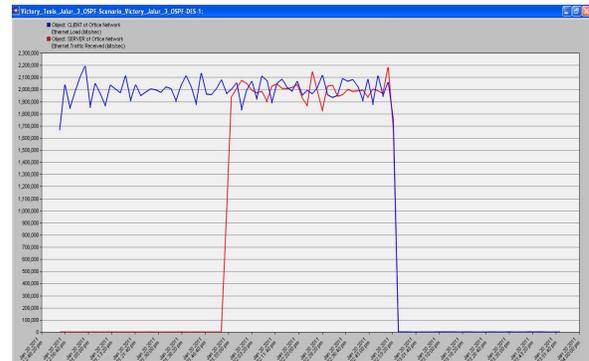
Namun data yang dilewatkan dari *client* (*received*) sampai ke *server* (*load*) dengan kapasitas data yang sama dari *maximal* 4.790 *bits* dan minimal 900 *bits* dengan waktu simulasi selama 2 jam



**Gambar 46. Overlaid Statistics Data Client (*received*) dan Server (*load*) Protocol OSPF Jalur 3 Duration 2 Jam**

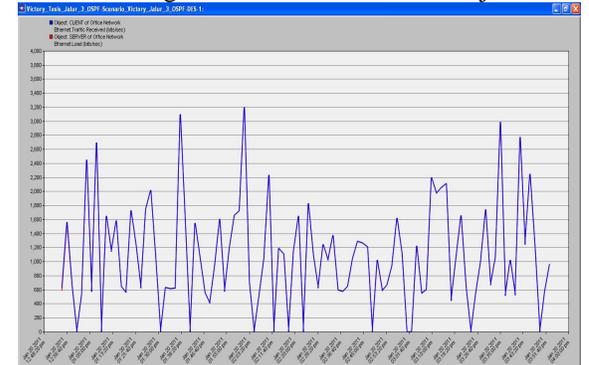
#### 5.9.4. Overlaid Statistics Data Client dan Server Protocol OSPF Jalur 3 Duration 3 Jam

Membandingkan tampilan data (*load*) yang bersumber dari *client* dan *server* menerima data (*traffic received (bit/sec)*) tersebut yang sebelumnya melalui *Router 1, Router 5, Router 6, Router 7, Router 8* dan *Router 2* data yang diterima sekitar 100% yaitu dari *maximal* data 2.200.000 *bits* dan *minimal* data 1.830.000 *bits* sedangkan data yang diterima *server* dari *maximal* 2.190 *bits* dan minimal 1.820.000 *bits*



**Gambar 47. Overlaid Statistics Client (*load*) dan Server (*received*) Protocol OSPF Jalur 3 Duration 3 Jam**

Namun data yang dilewatkan dari *client* (*received*) sampai ke *server* (*load*) dengan kapasitas data yang sama dari *maximal* 3.200 *bits* dan *minimal* 1.000 *bits* dengan waktu simulasi selama 3 jam



**Gambar 48. Overlaid Statistics Data Client (*received*) dan Server (*load*) Protocol OSPF Jalur 3 Duration 3 Jam**

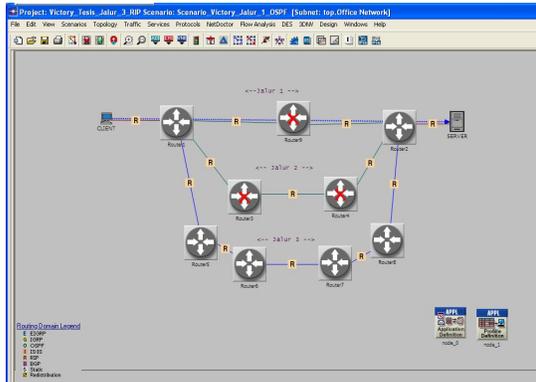
#### 5.10. Scenario Network Model Protocol RIP Jalur 3

*Scenario Network Model* Jalur 3 di gunakan untuk melewati data (*load*) secara keseluruhan (data *drop* pada *server*), Node pada jalur 3 terdiri atas :

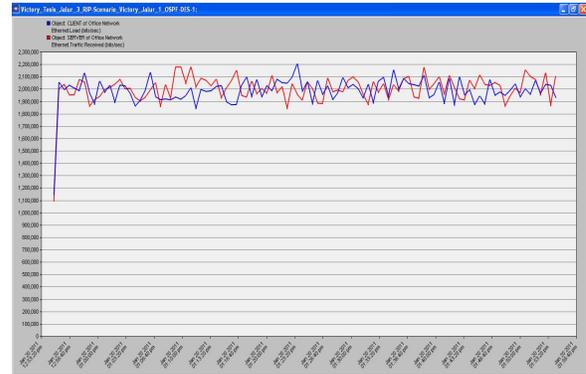
- a. *Client*
- b. *Router 1, Router 5, Router 6, Router 7, Router 8* dan *Router 2*
- c. *Server*

*Scenario Network Model* Jalur 3 di pisahkan agar dalam simulasi analisa output hanya menampilkan jalur 3 saja, jalur yang di pisahkan antara lain :

- a. Menonaktifkan jalur 1
- b. Dan menonaktifkan jalur 2



Gambar 49. Scenario Network Model Protocol RIP Jalur 3



Gambar 50. Overlaid Statistics Client (load) dan Server (received) Protocol RIP Jalur 3 Duration 1 Jam

### 5.10.1. Results Browser Routing Protocol RIP Jalur 3

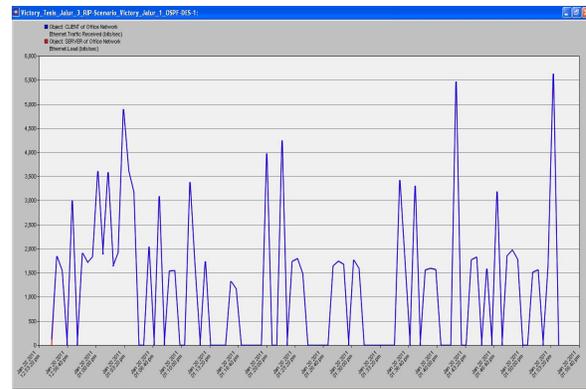
Merupakan tampilan hasil *input* atau output seluruh *node* yang telah di rancang dan di setting berdasarkan fungsi dan kebutuhan *Results Browser* Jalur 3.

Tabel 6. Results Browser Routing Protocol RIP Jalur 3

Jalur	Protocol	Stacket Statistics		Overlaid Statistics
		Client	Server	Client & Server
2	Load	√	√	√
2	Recieved	√	√	

### 5.10.2. Overlaid Statistics Data Client dan Server Protocol RIP Jalur 3 Duration 1 Jam

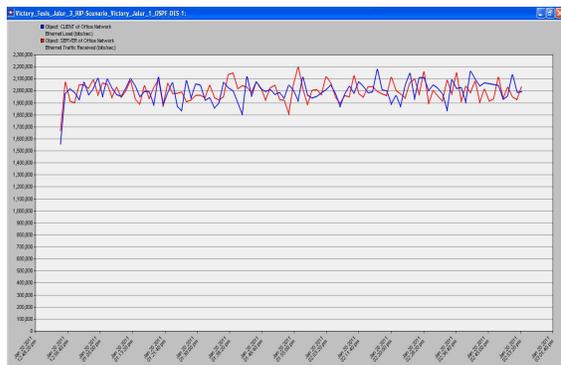
Membandingkan tampilan data (*load*) yang bersumber dari *client* dan *server* menerima data (*traffic received (bit/sec)*) tersebut yang sebelumnya melalui *Router 1, Router 5, Router 6, Router 7, Router 8* dan *Router 9* data yang diterima sekitar 50% yaitu dari *maximal* data 2.200.000 *bits* dan *minimal* data 1.850.000 *bits* sedangkan data yang diterima *server* dari *maximal* 2.190.000 *bits* dan *minimal* 1.850.000 *bits*.



Gambar 51. Overlaid Statistics Data Client (received) dan Server (load) Protocol RIP Jalur 3 Duration 1 Jam

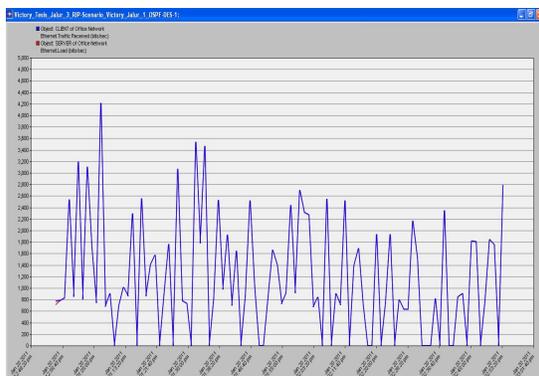
### 5.10.3. Overlaid Statistics Data Client dan Server Protocol RIP Jalur 3 Duration 2 Jam

Membandingkan tampilan data (*load*) yang bersumber dari *client* dan *server* menerima data (*traffic received (bit/sec)*) tersebut yang sebelumnya melalui *Router 1, Router 5, Router 6, Router 7, Router 8* dan *Router 2* data yang diterima sekitar 75% yaitu dari *maximal* data 2.090.000 *bits* dan *minimal* data 1.800.000 *bits* sedangkan data yang diterima *server* dari *maximal* 2.200.000 *bits* dan *minimal* 1.800.000 *bits*.



**Gambar 52. Overlaid Statistics Client (load) dan Server (received) Protocol RIP Jalur 3 Duration 2 Jam**

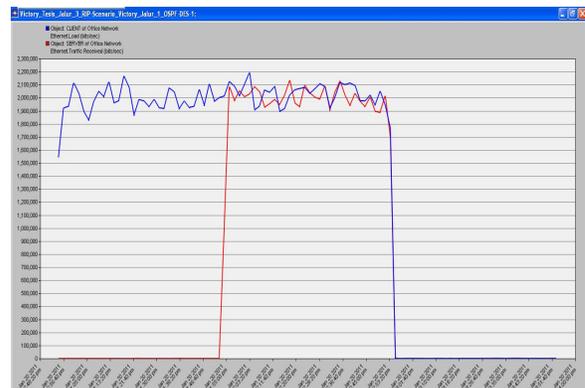
Namun data yang dilewatkan dari *client (received)* sampai ke *server (load)* dengan kapasitas data yang sama dari *maximal 4.200 bits* dan *minimal 800 bits* dengan waktu simulasi selama 2 jam.



**Gambar 53. Overlaid Statistics Client (load) dan Server (received) Protocol RIP Jalur 3 Duration 2 Jam**

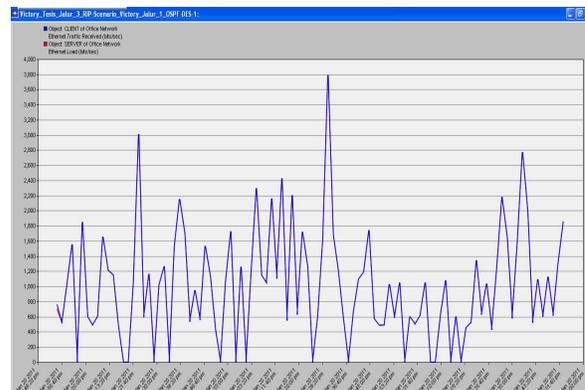
#### 5.10.4. Overlaid Statistics Data Client dan Server Protocol RIP Jalur 3 Duration 3 Jam

Membandingkan tampilan data (*load*) yang bersumber dari *client* dan *server* menerima data ( *traffic received (bit/sec)* ) tersebut yang sebelumnya melalui *Router 1, Router 5, Router 6, Router 7, Router 8* dan *Router 2* data yang diterima sekitar 100% yaitu dari *maximal data 2.090.000 bits* dan *minimal data 1.800.000 bits* sedangkan data yang diterima *server* dari *maximal 2.200.000 bits* dan *minimal 1.800.000 bits*.



**Gambar 54. Overlaid Statistics Client (load) dan Server (received) Protocol RIP Jalur 3 Duration 3 Jam**

Namun data yang dilewatkan dari *client (received)* sampai ke *server (load)* dengan kapasitas data yang sama dari *maximal 4.200 bits* dan *minimal 800 bits* dengan waktu simulasi selama 3 jam.



**Gambar 55. Overlaid Statistics Client (load) dan Server (received) Protocol RIP Jalur 3 Duration 3 Jam**

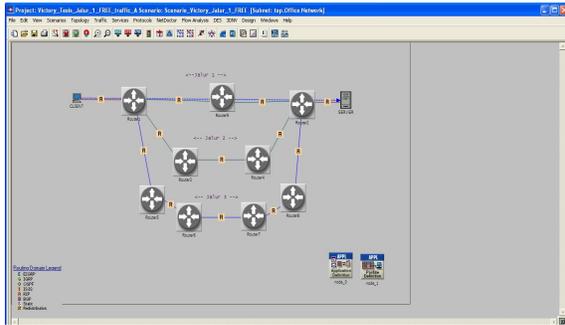
#### 5.11. Evaluasi

Berdasarkan *Results Browser Routing Protocol OSPF* dan *RIP* perlu dibuktikan dari *scenario network mode* akan mencari jalan terpendek untuk melewati data ke tujuan.

##### 5.11.1. Routing Protocol OSPF

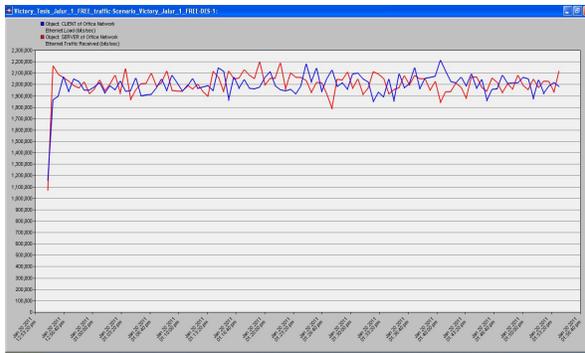
*Scenario Network Model* di gunakan untuk melewati data (*load*) dari *client* secara keseluruhan (data *drop*) pada *server*, dengan mengaktifkan jalur 1, jalur 2 dan jalur 3.





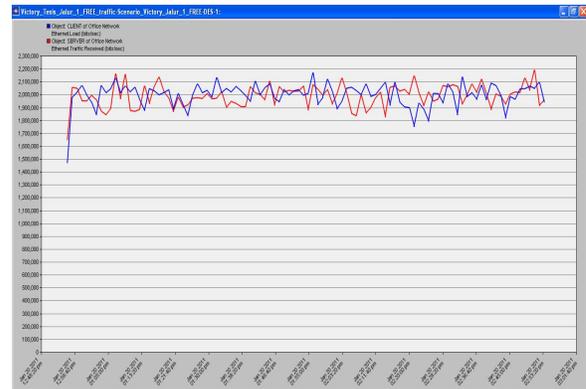
**Gambar 60. Scenario Network Model Protocol RIP**

1. *Routing Protocol RIP* dengan menganalisa waktu sampai data *client (load (bits/sec))* yaitu sekitar 50% dari *maximal data 2.200.000 bits* dan *minimal data 1.850.000 bits* sedangkan data yang diterima *server* dari *maximal 2.200.000 bits* dan *minimal 1.800.000 bits*. Melihat dari perpotongan pada waktu 1 jam pertama data yang yang dilewatkan tidak terlalu besar perbedaannya, Jadi data dilewatkan secara keseluruhan dari *node ke node* pada proses 1 jam pertama.



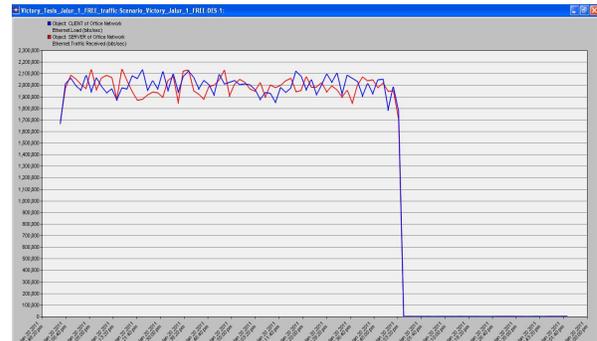
**Gambar 61. Overlaid Statistics Client (load) dan Server (received) Protocol RIP Duration 1 Jam**

2. *Routing Protocol RIP* dengan menganalisa waktu sampai data *client (load (bits/sec))* sekitar 75% yaitu dari *maximal data 2.190.000 bits* dan *minimal data 1.750.000 bits* sedangkan data yang diterima *server* dari *maximal 2.200.000 bits* dan *minimal 1.820.000 bits*. Simulasi 2 jam ini merupakan proses yang sama seperti 1 jam dapat dilihat dari data yang dilewatkan masih belum berakhir sehingga data masih dalam pemindahan dari *node ke node*.



**Gambar 62. Overlaid Statistics Client (load) dan Server (received) Protocol RIP Duration 2 Jam**

2. *Routing Protocol RIP* dengan menganalisa waktu sampai data *client (load (bits/sec))* sekitar 100% yaitu dari *maximal data 2.120.000 bits* dan *minimal data 1.890.000 bits* sedangkan data yang diterima *server* dari *maximal 2.150.000 bits* dan *minimal 1.850.000 bits*. Pada jam 02:55 data telah terlewatkan keseluruhan sehingga informasi dapat diterima seutuhnya.



**Gambar 63. Overlaid Statistics Client (load) dan Server (received) Protocol RIP Duration 3 Jam**

## 6. Kesimpulan dan Saran

### 6.1. Kesimpulan

Setelah dilakukan simulasi *scenario network model* dengan mengubah *routing protocol OSPF* dan *RIP* dapat disimpulkan :

1. Karakteristik *Routing protocol dengan OSPF* sebelum data dikirim, *protocol* menentukan jalur pengiriman yang terpendek dengan mengabaikan banyak *router* yang ada pada jalur tersebut.
2. Karakteristik *Routing protocol dengan RIP* menentukan jalur yang akan digunakan untuk

melewatkan data dengan mempertimbangkan jumlah *router* yang paling sedikit, hal ini dimungkinkan karena *protocol RIP* hanya dapat mengenal *router* tetangganya saja.

3. Simulasi yang dilakukan tidak dapat mengidentifikasi waktu hingga detik secara tepat yang merupakan perbandingan *routing protocol OSPF* dan *RIP* secara detail.

## 6.2. Saran

Setelah tim menyelesaikan tahapan akhir dari penelitian ini, tim menyadari masih banyak kekurangan-kekurangan yang mesti diperbaiki dan dipenuhi antara lain :

1. Pengujian *routing protocol* dengan *algoritma* yang sama.
2. *Scenario network model* dirancang dengan banyak *router* perjalur sama.
3. Hasil output yang akan di teliti lebih bervariasi agar lebih mendekati pada *object*.

## Referensi

- [1] Zainal Arifin, 2005, "Langkah Mudah Membangun Jaringan Komputer", Penerbit Andi.
- [2] Wendell Odom (2004:5), <http://www.digituck.com/pengertian-jaringan-komputer.html> dikutip 18 november 2010
- [3] The Third Wave - Alvin Toffler, 1991.
- [4] <http://ayucsg.blogspot.com/2009/10/pengertian-router.html>
- [5] Opnet.com
- [6] [http://en.wikipedia.org/wiki/Open\\_Shortest\\_Path\\_First](http://en.wikipedia.org/wiki/Open_Shortest_Path_First)
- [7] [http://www.ittelkom.ac.id/library/index.php?view=article&catid=10%3Ajaringan&id=527%3Aospf&option=com\\_content&Itemid=15](http://www.ittelkom.ac.id/library/index.php?view=article&catid=10%3Ajaringan&id=527%3Aospf&option=com_content&Itemid=15)
- [8] [http://nic.unud.ac.id/~lie\\_jasa/Artikel\\_reg\\_13.pdf](http://nic.unud.ac.id/~lie_jasa/Artikel_reg_13.pdf)
- [9] <http://win7dl.com/main/171424-opnet-modeler-140apl3.html>
- [10] <http://win7dl.com/search/mpls+nuggets.html>
- [11] [http://en.wikipedia.org/wiki/Dijkstra%27s\\_algorithm](http://en.wikipedia.org/wiki/Dijkstra%27s_algorithm)
- [12] <http://www.ta.trisakti.ac.id/TA/?q=node/200>
- [13] <http://ehibinitie.com/?p=28>
- [14] [http://asacreative.com/index.php?option=com\\_content&view=article&id=29:networking-lan-man-LAN&catid=60:tutorial&Itemid=16](http://asacreative.com/index.php?option=com_content&view=article&id=29:networking-lan-man-LAN&catid=60:tutorial&Itemid=16)
- [15] <http://id.wikipedia.org/wiki/Router>
- [16] [www.opnet.com/solutions/network\\_rd/odeler.html](http://www.opnet.com/solutions/network_rd/odeler.html)
- [17] Tommy Svensson dan Alex Popescu, Development of laboratory exercises based on OPNET Modeler, Blekinge Institute of Technology, Jul.