

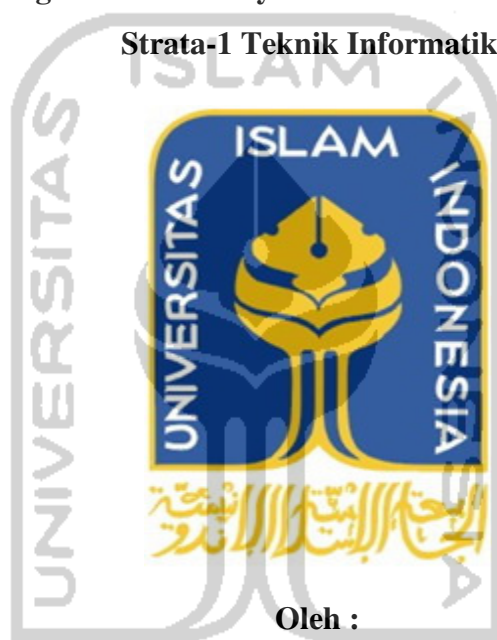
# **HALAMAN JUDUL**

## **ANALISIS KINERJA ROUTING PROTOCOL OSPF, BGP, DAN BGP MENGGUNAKAN POLICY**

### **LAPORAN TUGAS AKHIR**

**Diajukan Sebagai Salah Satu Syarat Untuk Memperoleh Gelar Sarjana**

**Strata-1 Teknik Informatika**



**Oleh :**

**Nama : Maulana Hidayat**

**No. Mahasiswa : 08523451**

**JURUSAN TEKNIK INFORMATIKA**

**FAKULTAS TEKNOLOGI INDUSTRI**

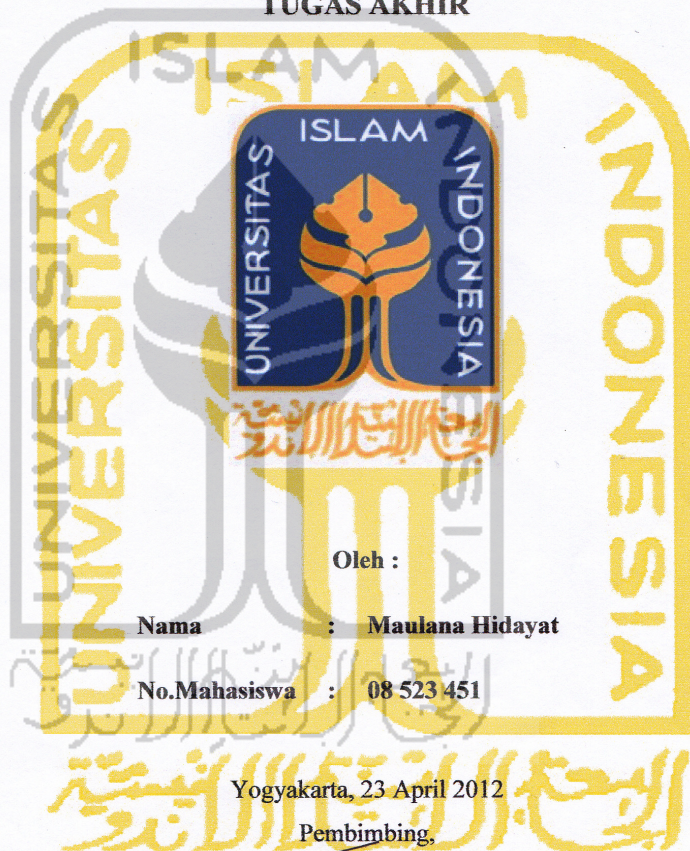
**UNIVERSITAS ISLAM INDONESIA**

**YOGYAKARTA**

**LEMBAR PENGESAHAN DOSEN PEMBIMBING**

**ANALISIS KINERJA ROUTING PROTOCOL OSPF, BGP, DAN  
BGP MENGGUNAKAN POLICY**

**TUGAS AKHIR**



Oleh :

Nama : Maulana Hidayat

No.Mahasiswa : 08 523 451

Yogyakarta, 23 April 2012

Pembimbing,

Syarif Hidayat ,S.Kom., M.I.T

## LEMBAR PERNYATAAN KEASLIAN HASIL TUGAS AKHIR

Saya yang bertandatangan di bawah ini,

**Nama : Maulana Hidayat**

**No. Mahasiswa : 08 523 451**

Menyatakan bahwa seluruh komponen dan isi dalam Laporan Tugas Akhir ini adalah hasil karya saya sendiri. Apabila di kemudian hari terbukti bahwa ada beberapa bagian dari karya ini adalah bukan hasil karya saya sendiri, maka saya siap menanggung risiko dan konsekuensi apapun.

Demikian pernyataan ini saya buat, semoga dapat dipergunakan sebagaimana mestinya.

Yogyakarta, 23 April 2012

  
Maulana Hidayat





# LEMBAR PENGESAHAN DOSEN PENGUJI

## ANALISIS KINERJA ROUTING PROTOCOL OSPF, BGP, DAN BGP MENGGUNAKAN POLICY

### TUGAS AKHIR

Oleh :

Nama : Maulana Hidayat

No. Mahasiswa : 08523451

Telah Dipertahankan di Depan Sidang Penguji sebagai Salah Satu Syarat  
untuk Memperoleh Gelar Sarjana Jurusan Teknik Informatika Fakultas  
Teknologi Industri Universitas Islam Indonesia

Yogyakarta, 23 April 2012

Tim Penguji,

Tim Penguji,

Syarif Hidayat S.Kom., M.I.T

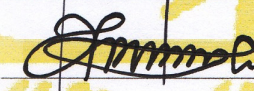
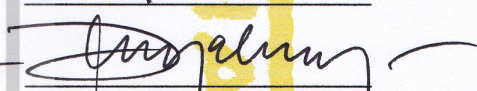
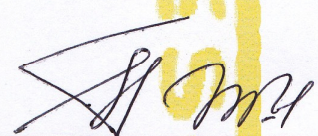
Ketua

Raden Teduh Dirgahayu Dr.S.T.,M.Sc.

Anggota 1

Hendrik S.T., M.Eng

Anggota 2

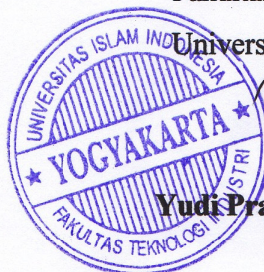


Mengetahui,

Ketua Jurusan Teknik Informatika

Fakultas Teknologi Industri

Universitas Islam Indonesia



Yudi Prayudi, S.Si., M.Kom.



## HALAMAN PERSEMBAHAN

*Tugas Akhir ini kudedikasikan dan kupersembahkan kepada...*

*Allah SWT, Tuhan Yang Maha Esa, yang hanya karena berkah dan limpahan rahmat-Nya lah tugas akhir ini dapat diselesaikan tepat pada waktunya.*

*Orang tuaku, saudara dan adikku tercinta juga seluruh keluarga besar yang turut memberikan dukungan.*

*Teman – teman seperjuangan di Laboratorium Terpadu, Teknik Informatika, VII. Juga semua teman dan pihak yang turut membantu dalam penyusunan tugas akhir ini.*

## HALAMAN MOTTO

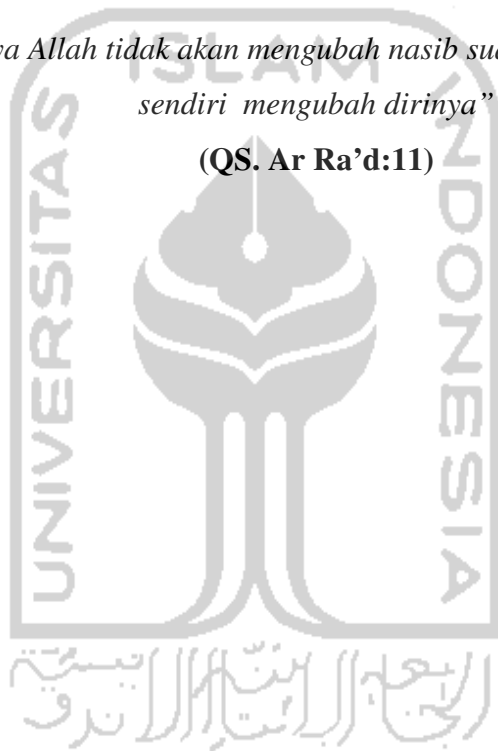
*“Selalu bersungguh – sungguh dan selalu diawali dengan niat hanya karena Allah semata dalam melakukan suatu usaha”*

*“Allah tidak membebani seseorang kecuali sepadan dengan kemampuannya...”*

**(QS. Al Baqarah:286)**

*“Sesungguhnya Allah tidak akan mengubah nasib suatu kaum sampai mereka sendiri mengubah dirinya”*

**(QS. Ar Ra’d:11)**





## KATA PENGANTAR



Alhamdulillah Rabbil'alamin. Puji dan syukur penulis panjatkan ke hadirat Allah SWT, Tuhan Semesta Alam, atas berkah, rahmat, karunia serta hidayah-Nya yang begitu besar kepada umat manusia. Juga tidak lupa pula shalawat serta salam yang senantiasa tercurah kepada Baginda Rasul, Muhammad SAW beserta para sahabat dan pengikutnya hingga akhir zaman, hingga tersusunlah tugas akhir dengan judul **“Analisis Kinerja Routing Protocol OSPF, BGP, dan BGP Menggunakan Policy”**.

Laporan tugas akhir ini disusun sebagai salah satu syarat untuk memperoleh gelar sarjana pada Jurusan Teknik Informatika, Fakultas Teknologi Industri, Universitas Islam Indonesia.

Penulisan dan penyelesaian tugas akhir ini tidak lepas dari saran, bimbingan, dukungan serta bantuan dari berbagai pihak. Untuk itu pada kesempatan ini penulis ingin mengucapkan terima kasih kepada:

1. Allah SWT yang selalu memberikan limpahan, berkah, rahmat, taufik dan hidayah-Nya kepada penulis.
2. Yang tercinta dan tersayang Ibunda (Dra. Normauliati ) dan Ayahanda (Drs. Suhandi), saudariku (Norlia Handayani) serta adikku (Rana Sausan), semoga Allah SWT membalas amal budinya dengan kasih sayang yang berlimpah.
3. Bapak Syarif Hidayat ,S.Kom., M.I.T, selaku pembimbing yang telah banyak meluangkan waktunya untuk sekedar berbagi obrolan ringan maupun berdiskusi selama penyusunan tugas akhir.
4. Bapak Yudi Prayudi, S.Si., M.Kom., selaku Ketua Jurusan, Jurusan Teknik Informatika, Fakultas Teknologi Industri, Universitas Islam Indonesia.
5. Sahabat-sahabatku tercinta di Lab Tutorial, Timur, Pandu, Amin, Lab GMM, dan juga teman – teman di Lab Sisjarkom 07, 08 dan 09 atas dukungan, bantuan, motivasi dan persahabatannya selama ini.

6. Teman – teman di komunitas Parlente.
7. Keluarga besar Informatika UII.
8. Serta semua pihak yang turut membantu baik itu secara langsung maupun tidak langsung hingga selesainya penyusunan tugas akhir ini. Hanya Allah SWT yang dapat membalas amal jariyah dan budi baik kalian semua.

Penulis sangat menyadari keterbatasan pengetahuan dan kemampuan penulis. Oleh karena itu jika terdapat banyak kesalahan dalam penulisan laporan tugas akhir ini, penulis menerima kritik dan sarannya demi kebenaran dan kegunaan informasi yang penulis berikan kelak.

Akhir kata, semoga tugas akhir ini dapat memberikan suatu manfaat yang sebesar-besarnya baik bagi pribadi penulis sendiri maupun bagi kita semua. Amin.

Yogyakarta, 2012

Penulis,

Maulana Hidayat





## SARI

Setiap *Internet Service Provider* (ISP), biasanya memiliki kebijakan sendiri dalam mengelola jaringannya. Dikenal istilah *Autonomous System* (AS) untuk mendefinisikan seperangkat *router* yang berada pada satu kebijakan tunggal. Untuk memperkenalkan jaringan yang ada pada suatu AS ke AS lainnya digunakan *Eksterior Gateway Protocol* (EGP). Satu – satunya contoh EGP yang digunakan saat ini yaitu *Border Gateway Protocol* (BGP). BGP memiliki beberapa kelebihan dan kekurangan dibandingkan *Interior Gateway Protocol* (IGP) misalnya saja *Open Shortest Path First* (OSPF). IGP digunakan untuk keperluan *routing* pada satu AS yang sama. Penggunaan BGP dan OSPF haruslah tepat agar jaringan yang dikelola menjadi optimal. Tidak menutup kemungkinan jika administrator suatu jaringan tidak memahami kinerja sebenarnya dari dua protokol ini, maka jaringan yang dikelola menjadi tidak optimal dan bahkan bisa mengakibatkan gagalnya jalur tertentu. BGP juga memiliki beberapa atribut yang dapat digunakan sebagai acuan dalam menentukan *policy* yang diterapkan pada suatu *router*. Penerapan *policy* akan mempengaruhi kinerja dari keseluruhan jaringan. Tujuan dari penelitian ini adalah untuk mengetahui bagaimana kinerja *routing protocol* BGP dibandingkan OSPF dan juga penggunaan *policy* pada BGP.

Pada tugas akhir ini penulis menggunakan dua metodologi penelitian, yaitu studi pustaka dan implementasi dan analisis sistem. Pada studi pustaka, penulis menggunakan beberapa literatur dari internet dan juga media cetak untuk mengumpulkan informasi terkait tugas akhir. Sedangkan pada implementasi dan analisis sistem, penulis menyusun beberapa batasan implementasi dan juga langkah pengerjaan. Batasan implementasi yang penulis terapkan yaitu penggunaan simulator OPNET Modeler, penggunaan beberapa skenario, konfigurasi *routing protocol* secara *default*, dan banyaknya penggunaan perangkat pada simulasi ini. Sedangkan langkah pengerjaannya yaitu dimulai dari melakukan analisis kebutuhan sistem, instalasi dan konfigurasi sistem, desain simulasi, konfigurasi simulasi, pengujian sistem, dan pengambilan kesimpulan.

Dari hasil penelitian pada tugas akhir menggunakan OPNET Modeler ini, penulis dapat simpulkan bahwa penerapan BGP dengan menggunakan satu buah *policy* tidak terlalu menimbulkan perubahan yang berarti pada kinerja jaringan. Perubahan yang terlihat yaitu pada beberapa perbandingan pengamatan antara OSPF dan BGP (juga BGP dengan *policy*).

Kata kunci : *internet service provider, autonomous system, eksterior gateway protocol, border gateway protocol, interior gateway protocol, Open Shortest Path First, policy, routing protocol, OPNET Modeler.*

## TAKARIR



<i>Advertise</i>	Mengiklankan
<i>Backbone</i>	Jalur inti/penopang
<i>Backup</i>	Cadangan
<i>Bandwidth</i>	Lebar pita
<i>Broadcast</i>	Disebarkan ke semua
<i>Client</i>	Pihak yang meminta layanan
<i>Collision</i>	Tabrakan
<i>Decode</i>	Mengembalikan format asal
<i>Encode</i>	Mengubah jadi format tertentu
<i>File</i>	Data
<i>Flooding</i>	Membanjiri trafik
<i>Hop count</i>	Hitungan lompatan
<i>Hop-by-hop</i>	Lompatan demi lompatan
<i>Layer</i>	Lapisan
<i>Loop</i>	Berulang – ulang
<i>Neighbor</i>	Tetangga
<i>Node</i>	Perangkat
<i>Peer</i>	Rekan
<i>Policy</i>	Kebijakan
<i>Printer sharing</i>	Berbagi fungsi printer di jaringan
<i>Proprietary</i>	Hak milik
<i>Remote</i>	Akses jarak jauh
<i>Router</i>	Perangkat untuk merutekan data
<i>Routing</i>	Proses merutekan
<i>Routing protocol</i>	Aturan routing
<i>Scalable</i>	Dapat diperbesar/diperluas
<i>Server</i>	Penyedia layanan
<i>Storage sharing</i>	Berbagi tempat penyimpanan
<i>Summarize</i>	Menyingkat
<i>Video conference</i>	Video beberapa pihak langsung
<i>Workstation</i>	Perangkat yang berfungsi sebagai klien



## DAFTAR ISI

<b>LEMBAR PENGESAHAN DOSEN PEMBIMBING .....</b>	<b>ii</b>
<b>LEMBAR PERNYATAAN KEASLIAN HASIL TUGAS AKHIR ....</b>	<b>iii</b>
<b>LEMBAR PENGESAHAN DOSEN PENGUJI .....</b>	<b>iv</b>
<b>HALAMAN PERSEMBAHAN .....</b>	<b>v</b>
<b>HALAMAN MOTTO .....</b>	<b>vi</b>
<b>KATA PENGANTAR.....</b>	<b>vii</b>
<b>SARI .....</b>	<b>ix</b>
<b>TAKARIR.....</b>	<b>x</b>
<b>DAFTAR ISI.....</b>	<b>xi</b>
<b>DAFTAR GAMBAR.....</b>	<b>xiii</b>
<b>DAFTAR TABEL .....</b>	<b>xvi</b>
<b>BAB I PENDAHULUAN.....</b>	<b>1</b>
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	2
1.3 Batasan Masalah.....	2
1.4 Tujuan Penelitian .....	2
1.5 Manfaat Penelitian .....	3
1.6 Metodologi Penelitian .....	3
1.6.1 Studi Pustaka.....	3
1.6.2 Implementasi dan Analisis Sistem .....	3
1.7 Sistematika Penulisan .....	5
<b>BAB II LANDASAN TEORI .....</b>	<b>7</b>
2.1 Jaringan Komputer .....	7
2.1.1 Topologi Fisik Jaringan Komputer .....	7
2.1.2 Topologi Logik Jaringan Komputer.....	10
2.1.3 Jenis Jaringan Komputer Berdasarkan Area .....	10
2.2 OSI dan TCP/IP .....	11
2.3 <i>Routing Protocol</i> .....	14
2.3.1 <i>OSPF (Open Shortest Path First)</i> .....	15
2.3.1 <i>BGP (Border Gateway Protocol)</i> .....	17
2.4 Simulator OPNET MODELER 14.0.....	21
2.4.1 <i>Project Editor</i> .....	22
2.4.2 <i>Node Editor</i> .....	23
2.4.3 <i>Process Model Editor</i> .....	23
2.4.4 <i>Link Model Editor</i> .....	24
2.4.5 <i>Path Editor</i> .....	25

2.4.6	<i>Simulation Sequence Editor</i> .....	25
<b>BAB III METODOLOGI PENELITIAN .....</b>		<b>26</b>
3.1	Analisis Kebutuhan Sistem .....	26
3.1.1	Perangkat Keras yang Dibutuhkan.....	26
3.1.2	Perangkat Lunak yang Dibutuhkan.....	26
3.2	Instalasi dan Konfigurasi Sistem.....	27
3.3	Desain Sistem.....	28
3.3.1	Skenario OSPF_Simple.....	29
3.3.2	Skenario BGP_Simple .....	29
3.3.3	Skenario BGP_Policy .....	29
3.3.4	Skenario OSPF_Http, BGP_Http, dan BGP_Policy_Http ....	30
3.3.5	Skenario OSPF_Voice, BGP_Voice, dan BGP_Policy_Voice. .....	30
3.3.6	Skenario OSPF_Video, BGP_Video, dan BGP_Policy_Video .....	30
3.4	Konfigurasi Simulasi.....	31
3.4.1	Skenario OSPF_Http.....	31
3.4.2	Skenario BGP_Http.....	42
3.4.3	Skenario BGP_Policy_Http .....	47
3.4.4	Skenario OSPF_Voice, BGP_Voice, dan BGP_Policy_Voice. .....	50
3.4.5	Skenario OSPF_Video, BGP_Video, dan BGP_Policy_Video .....	53
3.4.6	Skenario OSPF_Simple, BGP_Simple, dan BGP_Policy..... .....	56
3.5	Pengujian Sistem.....	59
<b>BAB IV HASIL DAN PEMBAHASAN .....</b>		<b>60</b>
4.1	Analisis dan Evaluasi Hasil Uji Coba.....	60
4.1.1	Perbandingan Skenario OSPF_Simple, BGP_Simple, dan BGP_Policy.....	60
4.1.2	Perbandingan Skenario OSPF_Http, BGP_Http, dan BGP_Policy_Http.....	64
4.1.3	Perbandingan Skenario OSPF_Voice, BGP_Voice, dan BGP_Policy_Voice .....	67
4.1.4	Perbandingan Skenario OSPF_Video, BGP_Video, dan BGP_Policy_Video.....	69
<b>BAB V PENUTUP.....</b>		<b>71</b>
5.1	Kesimpulan .....	71
5.2	Saran.....	72
<b>DAFTAR PUSTAKA .....</b>		<b>73</b>

## DAFTAR GAMBAR

Gambar 2.1 Topologi Bus .....	7
Gambar 2.2 Topologi Ring .....	8
Gambar 2.3 Topologi Star.....	9
Gambar 2.4 Topologi Mesh (webune, 2012) .....	9
Gambar 2.5 Perbandingan OSI dan TCP/IP (CCNA, 2010).....	14
Gambar 2.6 Contoh Topologi Sederhana BGP .....	18
Gambar 2.7 Workflow OPNET Modeler (Svensson,2003) .....	22
Gambar 2.8 Project Editor Pada OPNET .....	22
Gambar 2.9 Node Model.....	23
Gambar 2.10 Process Model .....	24
Gambar 2.11 Link Model Editor.....	24
Gambar 2.12 Path Editor (Svensson, 2003).....	25
Gambar 2.13 Editor Simulasi OPNET.....	25
Gambar 3.1 Desain Sistem.....	28
Gambar 3.2 Pembuatan Project Baru.....	31
Gambar 3.3 Topologi Project Kinerja_BGP .....	33
Gambar 3.4 Konfigurasi Address Family pada OSPF Parameters .....	33
Gambar 3.5 Konfigurasi Redistribusi Pada OSPF.....	34
Gambar 3.6 Report Pada Tabel Routing OSPF.....	34
Gambar 3.7 Aktivasi Interface Loopback .....	35
Gambar 3.8 Konfigurasi Simulasi.....	36
Gambar 3.9 Aktivasi OSPF Pada Interface.....	38
Gambar 3.10 Konfigurasi Layanan Web Service .....	38
Gambar 3.11 Application dan Profile Config .....	39
Gambar 3.12 Konfigurasi Profile untuk Web Service .....	39
Gambar 3.13 Konfigurasi Permintaan Web Service Pada Client .....	40
Gambar 3.14 Parameter Pengamatan Global OSPF_Http.....	41
Gambar 3.15 Parameter Pengamatan Node OSPF_Http.....	41

Gambar 3.16 Duplikasi Skenario .....	42
Gambar 3.17 Konfigurasi IPv4 Pada BGP.....	43
Gambar 3.18 Distribusi BGP Melalui OSPF .....	43
Gambar 3.19 Konfigurasi AS Number.....	44
Gambar 3.20 Aktivasi Routing Protocol BGP .....	45
Gambar 3.21 Konfigurasi Pada BGP Neighbor .....	47
Gambar 3.22 Jalur Dari Server Menuju Client Pada BGP_Simple .....	47
Gambar 3.23 Modifikasi Atribut Local Preference.....	49
Gambar 3.24 Implementasi Policy Pada Neighbor BGP .....	49
Gambar 3.25 Konfigurasi Layanan VOIP.....	50
Gambar 3.26 Konfigurasi Profile Untuk VOIP.....	51
Gambar 3.27 Konfigurasi Permintaan VOIP Pada Klien.....	52
Gambar 3.28 Parameter Pengamatan Global VOIP.....	52
Gambar 3.29 Parameter Pengamatan Node VOIP .....	53
Gambar 3.30 Konfigurasi Layanan Video Conference.....	54
Gambar 3.31 Konfigurasi Profile Untuk Video Conference.....	54
Gambar 3.32 Konfigurasi Permintaan Video Conference Pada Klien.....	55
Gambar 3.33 Parameter Pengamatan Global Video Conference .....	56
Gambar 3.34 Parameter Pengamatan Node Video Conference .....	56
Gambar 3.35 Konfigurasi Penonaktifan Layanan Pada Server.....	57
Gambar 3.36 Konfigurasi Penonaktifan Layanan Pada Klien .....	57
Gambar 3.37 Pengamatan Trafik OSPF.....	58
Gambar 3.38 Pengamatan Trafik BGP.....	58
Gambar 3.39 Pengamatan Penggunaan Resource CPU .....	59
Gambar 4.1 Perbandingan Trafik Protokol Routing Terkirim.....	60
Gambar 4.2 Trafik OSPF Terkirim .....	61
Gambar 4.3 Perbandingan Trafik BGP_Simple dan BGP_Policy .....	62
Gambar 4.4 Perbandingan Persentase CPU Utilization .....	62
Gambar 4.5 Perbandingan Rata-Rata Waktu Respon Halaman Web .....	65
Gambar 4.6 Perbandingan Trafik Web yang Diterima Klien .....	66
Gambar 4.7 Perbandingan Variasi Delay pada VOIP .....	67

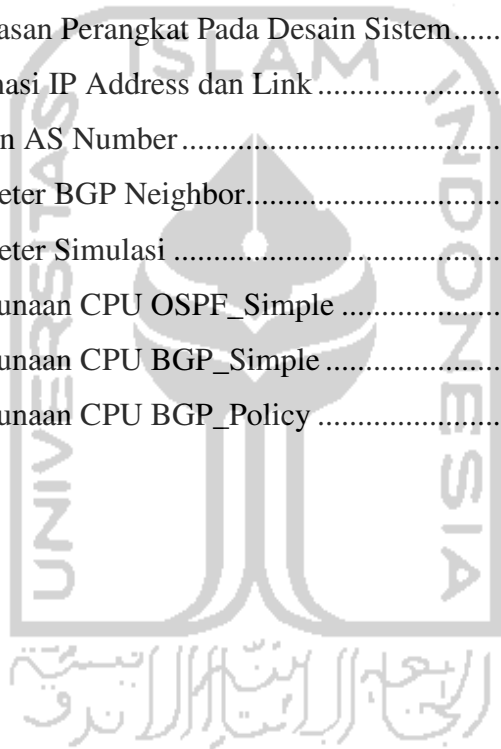


Gambar 4.8 Perbandingan Trafik VOIP yang Diterima Klien..... 68  
Gambar 4.9 Perbandingan Variasi Delay Pada Video Conference..... 69  
Gambar 4.10 Perbandingan Trafik Video yang Diterima Klien ..... 70



## DAFTAR TABEL

Tabel 2.1 Cakupan Area Jaringan Komputer (Sofana, 2008:5) .....	11
Tabel 2.2 Penjelasan Model OSI (CCNA, 2010).....	12
Tabel 2.3 Penjelasan Model TCP/IP (Sofana, 2008:89) .....	13
Tabel 2.4 Penjelasan Cost Dari Beberapa Interface (CCNA, 2010).....	16
Tabel 2.5 BGP Message (Rafiudin, 2004:36) .....	19
Tabel 2.6 Atribut Well-Known BGP (White, et. al., 2004) .....	20
Tabel 3.1 Penjelasan Perangkat Pada Desain Sistem.....	32
Tabel 3.2 Informasi IP Address dan Link .....	36
Tabel 3.3 Rincian AS Number .....	44
Tabel 3.4 Parameter BGP Neighbor.....	45
Tabel 3.5 Parameter Simulasi .....	59
Tabel 4.1 Penggunaan CPU OSPF_Simple .....	63
Tabel 4.2 Penggunaan CPU BGP_Simple .....	63
Tabel 4.3 Penggunaan CPU BGP_Policy .....	64



# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Setiap *Internet Service Provider* (ISP) biasanya memiliki kebijakan sendiri dalam mengelola semua jaringan yang ada dibawah mereka. Dikenal juga istilah AS yang merupakan singkatan dari *Autonomous System* yang digunakan oleh ISP untuk mewakili kebijakan yang mereka terapkan pada jaringan mereka. “AS didefinisikan sebagai satu set *router* yang berada dibawah pengawasan administrasi tunggal” (Rafiudin, 2004:34). Setiap AS biasanya mengimplementasikan *routing protocol* yang berbeda – beda untuk menunjang proses *routing* antar *router*. Ada dua konsep *routing protocol* yang membentuk internet saat ini, yaitu *Eksterior Gateway Protocol* (EGP) dan *Interior Gateway Protocol* (IGP). Penggunaan IGP hanya dapat diimplementasikan pada satu domain *routing* yang sama, atau bisa juga disebut satu AS yang sama. Misalnya saja keperluan *routing* antar divisi dari suatu organisasi. Penggunaan IGP tidak dapat diimplementasikan antar AS yang berbeda.

Untuk memperkenalkan jaringan yang ada pada suatu AS ke AS lainnya digunakan EGP. “*Border Gateway Protocol* (BGP) merupakan satu – satunya contoh EGP yang berjalan saat ini dan merupakan *routing protocol* yang digunakan oleh internet” (CCNA, 2010). BGP digunakan sebagai *backbone* nya internet, yang digunakan antar AS bukan lagi pada satu AS. BGP memiliki beberapa kelebihan dan kekurangan dibandingkan protokol IGP lainnya misalnya saja *Open Shortest Path First* (OSPF). OSPF dijadikan acuan karena OSPF merupakan *routing protocol* IGP yang paling sering digunakan oleh administrator jaringan pada AS mereka. Penggunaan BGP dan OSPF haruslah tepat agar jaringan yang dikelola menjadi optimal. Tidak menutup kemungkinan jika administrator suatu jaringan tidak memahami kinerja dari dua *routing protocol*

ini, maka jaringan yang dikelola menjadi tidak optimal dan bahkan bisa mengakibatkan gagalnya jalur tertentu.

Sama seperti *routing protocol* lainnya, BGP juga memiliki beberapa atribut yang dapat digunakan sebagai acuan dalam menentukan *policy* yang diterapkan pada suatu *router*. Penerapan *policy* yang tepat tentunya akan mempengaruhi kinerja dari keseluruhan jaringan. Sebaliknya, jika administrator jaringan tidak memahami atribut dari *routing protocol* yang ia pakai, bisa jadi kinerja dari *routing protocol* tidak sesuai dengan harapan dan tidak memaksimalkan kinerja dari jaringan yang dikelola.

## 1.2 Rumusan Masalah

Berdasarkan latar belakang di atas maka didapat suatu rumusan masalah yaitu bagaimana sebenarnya kinerja dari *routing protocol* BGP, baik itu dengan diterapkannya BGP *policy* maupun dibandingkan dengan OSPF.

## 1.3 Batasan Masalah

Batasan masalah yang penulis kemukakan dalam tugas akhir ini adalah:

- a. Parameter yang diamati yaitu trafik *routing protocol*, penggunaan *resource* CPU pada *router*, dan pengamatan kinerja *routing protocol* dengan menggunakan beberapa aplikasi seperti VOIP, *video conference* dan aplikasi website.
- b. Ketidakakuratan pengukuran yang diakibatkan oleh simulator OPNET tidak dibahas dalam tugas akhir ini.

## 1.4 Tujuan Penelitian

Tujuan dari penelitian ini adalah untuk mengetahui bagaimana kinerja *routing protocol* BGP dibandingkan OSPF dan juga penggunaan *policy* pada BGP.



## 1.5 Manfaat Penelitian

Melalui pengimplementasian BGP, OSPF, dan juga BGP *policy* pada simulasi ini, diharapkan pihak – pihak yang menggunakan maupun ingin mempelajari *routing protocol* OSPF dan BGP, mendapatkan gambaran secara umum baik itu tentang trafik *routing protocol*, penggunaan *resource CPU* pada *router*, *throughput* pada *client*, dan kinerja *routing protocol* BGP, BGP dengan *policy*, dan OSPF dengan perbandingan menggunakan beberapa aplikasi seperti VOIP, *video conference* dan aplikasi website.

## 1.6 Metodologi Penelitian

Pada tugas akhir ini, penulis menggunakan beberapa metodologi yang digunakan sebagai acuan dalam pengerjaan, yaitu :

### 1.6.1 Studi Pustaka

Studi pustaka digunakan untuk mendapatkan informasi tambahan yang digunakan sebagai acuan dalam penelitian ini. Pada tahap ini penulis menggunakan beberapa literatur dari internet juga media cetak lainnya untuk mengetahui konsep dasar dari beberapa *routing protocol*, juga untuk mempelajari cara menggunakan software OPNET Modeler.

### 1.6.2 Implementasi dan Analisis Sistem

Pada pembuatan simulasi sistem, penulis menerapkan beberapa batasan, yaitu :

- a. Implementasi dilakukan dengan simulasi menggunakan program simulator OPNET Modeler 14.0 versi edukasi.
- b. Implementasi dilakukan dengan menggunakan beberapa skenario.
- c. *Router* yang digunakan sebanyak 9 buah, *server* 1 buah, dan LAN sebanyak 1 buah.
- d. Konfigurasi *routing protocol* OSPF dan BGP dilakukan dengan beberapa atribut yang diatur secara *default*, tidak secara mendalam.

- e. Skenario dengan menggunakan beberapa layanan hanya sebatas menggunakan layanan Http, VOIP, dan *video conference*.

Adapun mengenai langkah pengerjaan, penulis menyusun beberapa langkah pengerjaan berdasarkan hasil dari studi pustaka. Langkah – langkah pengerjaannya yaitu :

- a. Analisis Kebutuhan Sistem

Tahapan awal yang dilakukan yaitu melakukan analisis dan identifikasi mengenai perangkat lunak, dan hal – hal apa saja yang dibutuhkan dalam penelitian mengenai kinerja OSPF, BGP, dan BGP menggunakan *policy*.

- b. Instalasi dan Konfigurasi Sistem

Setelah tahapan awal selesai, maka penulis mulai melakukan instalasi OPNET Modeler 14.0 versi edukasi yang digunakan sebagai simulator dari keseluruhan topologi jaringan nantinya.

- c. Desain Simulasi

Pada tahap ini penulis merancang desain topologi jaringan yang akan dibuat nantinya, juga skenario dari penelitian kinerja OSPF, BGP, dan BGP menggunakan *policy*.

- d. Konfigurasi Simulasi

Setelah desain simulasi selesai, penulis akan membuat *project* dengan beberapa skenario sesuai dengan desain simulasi, kemudian melakukan konfigurasi pada objek yang ada.

- e. Pengujian Simulasi

Setelah konfigurasi simulasi selesai, penulis akan menjalankan simulasi dan melihat apakah ada proses yang *error* atau jalannya simulasi yang tidak sesuai dengan skenario yang telah dibuat.

- f. Pengambilan Kesimpulan

Tahap yang terakhir yaitu melakukan analisa pada statistik data yang berupa grafik, dan menuangkannya dalam bentuk laporan sehingga

nantinya akan didapatkan hasil dari beberapa skenario yang sudah diatur sebelumnya.

## 1.7 Sistematika Penulisan

Sistematika penulisan yang digunakan dalam pembuatan laporan tugas akhir ini adalah sebagai berikut:

### BAB I Pendahuluan

Pada bab ini penulis membahas tentang latar belakang masalah, batasan masalah, rumusan masalah, tujuan penelitian, manfaat dari penelitian dan metodologi penelitian yang digunakan dalam pelaksanaan tugas akhir mengenai analisis kinerja *routing protocol* OSPF, BGP, dan BGP menggunakan *policy*.

### BAB II Landasan Teori

Pada bagian ini berisi tentang dasar - dasar teori yang berhubungan dengan penelitian mengenai analisis kinerja *routing protocol* OSPF, BGP, dan BGP menggunakan *policy*.

### BAB III Metodologi

Pada bab ini akan dibahas analisis sistem yang dibutuhkan, instalasi sistem, desain simulasi, konfigurasi simulasi dan uji coba dari simulasi sistem yang telah dibuat.

### BAB IV Hasil dan Pembahasan

Pada bagian ini memuat tentang analisis dari hasil simulasi sistem, yang nantinya akan mengarah pada analisis statistik dari tiap – tiap bagian yang diamati, juga perbandingan antar skenario.

## BAB V Penutup

Pada bagian ini, penulis akan membahas kesimpulan dari hasil yang didapat pada bab sebelumnya. Bab ini juga memuat saran yang ditujukan bagi mahasiswa pada khususnya dan pembaca pada umumnya yang berminat untuk melanjutkan penelitian mengenai analisis kinerja *routing protocol* OSPF, BGP, dan BGP menggunakan *policy*. Penulis juga memaparkan beberapa keterbatasan pembahasan dalam penelitian ini.





## BAB II

### LANDASAN TEORI

#### 2.1 Jaringan Komputer

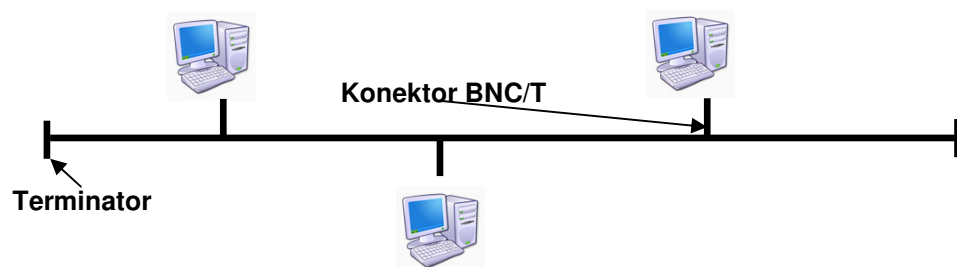
Jaringan komputer merupakan istilah yang dipakai untuk mendefinisikan kumpulan dari dua komputer atau lebih yang saling terhubung melalui media tertentu dengan menggunakan aturan – aturan tertentu. Media yang digunakan dapat berupa media fisik berupa kabel maupun gelombang radio, infrared, dan satelit.

##### 2.1.1 Topologi Fisik Jaringan Komputer

Pada saat implementasi di lapangan, terdapat beberapa topologi fisik yang sering digunakan. Topologi fisik jaringan komputer disini merupakan konfigurasi atau desain dari perangkat maupun media yang membentuk jaringan. Berikut merupakan beberapa contoh dari topologi jaringan komputer :

a. Bus

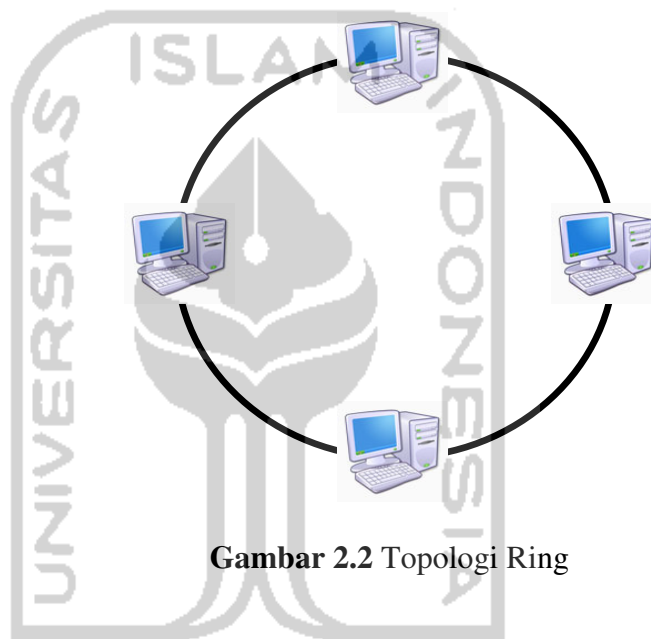
Pada topologi ini, beberapa komputer dihubungkan secara berantai melalui kabel tunggal. Media yang digunakan yaitu kabel coaxial, sedangkan konektor penghubungnya menggunakan konektor BNC/T. Kelemahan utama dari topologi ini yaitu adanya *collision* (tabrakan antar paket data) dan bila salah satu komputer mati maka akan mempengaruhi yang lainnya. Contoh topologi bus dapat dilihat pada gambar 2.1.



**Gambar 2.1** Topologi Bus

b. Ring

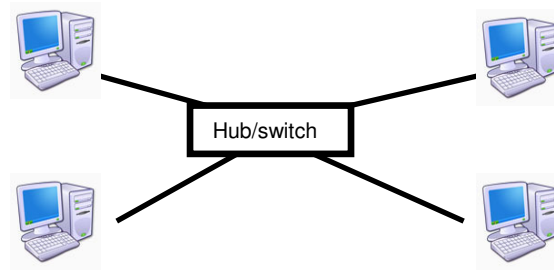
“Topologi ini mirip dengan topologi bus, bedanya topologi ring ujungnya saling terhubung” (Wahidin, 2007:11). Pada topologi ini, data mengalir searah sehingga tabrakan data dapat dihindari. Kelemahannya yaitu bila salah satu komputer mati, maka akan mempengaruhi komputer lainnya. Contoh topologi ring dapat dilihat pada gambar 2.2.



**Gambar 2.2** Topologi Ring

c. Star

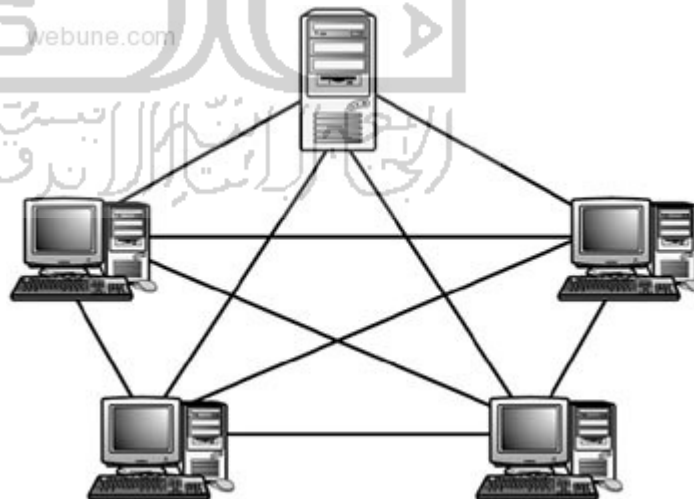
Star merupakan topologi yang populer digunakan saat ini. Pada topologi ini tiap komputer terhubung secara langsung melalui perangkat perantara berupa *hub/switch*. Pada *hub* masih terdapat *collision* dikarenakan penggunaan *hub* yang sifatnya melakukan *broadcast* paket data ke semua *node* beda halnya dengan *switch*. Penggunaan *switch* digunakan untuk meminimalisir *collision* yang terjadi karena paket data dikirimkan secara *point-to-point*. Contoh topologi star dapat dilihat pada gambar 2.3.



**Gambar 2.3** Topologi Star

d. Mesh

Topologi mesh merupakan topologi rumit yang biasanya digunakan untuk suatu jaringan yang sangat penting keberadaannya. Topologi ini disebut juga hubungan *point-to-point* dikarenakan setiap *node* saling terhubung dengan *node* lainnya. Kelebihan utama dari topologi ini yaitu apabila salah satu jalur mati maka akan tersedia *backup* jalur lainnya. Topologi mesh banyak diterapkan pada WAN, juga internet. Contoh topologi mesh dapat dilihat pada gambar 2.4.



**Gambar 2.4** Topologi Mesh (webune, 2012)

## 2.1.2 Topologi Logik Jaringan Komputer

### a. Client Server

Istilah *client server* mengacu pada suatu topologi jaringan dengan keberadaan satu atau lebih perangkat yang bertindak sebagai *server* dan juga satu atau lebih perangkat yang bertindak sebagai *client*. Tugas *server* yaitu melayani permintaan paket data dari *client*.

### b. Peer to Peer

Jaringan *peer to peer* merupakan jaringan yang terdiri dari dua komputer atau lebih yang saling terhubung melalui media tertentu. Pada *peer to peer*, tidak terdapat istilah komputer *client* ataupun *server*. Semua komputer yang saling terhubung dalam jaringan ini dapat berfungsi sebagai *server* dan juga *client* secara bersamaan.

### c. Host Terminal

“Topologi logik host terminal terdiri dari satu atau lebih *server* yang dihubungkan dalam suatu dumb terminal” (Sisjarkom, 2009). Komputer *server* bertindak sebagai perangkat yang mengolah data, sedangkan terminal hanyalah sebuah monitor yang terhubung dengan menggunakan kabel RS-232. Jadi semua proses dari terminal akan dikerjakan di komputer *server* yang tentu saja membutuhkan spesifikasi perangkat keras yang tinggi agar dapat menjaga performa dari terminal – terminal yang terhubung.

## 2.1.3 Jenis Jaringan Komputer Berdasarkan Area

### a. LAN

*Local Area Network* (LAN) merupakan jaringan lokal yang terdiri dari dua komputer atau lebih yang saling terhubung pada jarak yang relatif lebih dekat. LAN biasa digunakan untuk keperluan pribadi dalam skala kecil, misalnya penggunaan *printer sharing*, *storage sharing*, dan lain lain.

## b. MAN

*Metropolitan Area Network* (MAN) definisinya hampir sama dengan LAN namun berbeda dalam hal cakupan wilayah. Cakupan wilayah MAN lebih luas dibandingkan LAN. Cakupan MAN bisa antar kantor yang berbeda, antar kabupaten, bahkan provinsi.

## c. WAN

*Wide Area Network* (WAN) merupakan jaringan komputer yang hampir sama dengan LAN dan MAN namun memiliki skala yang lebih luas lagi dibandingkan dengan MAN. Cakupan WAN meliputi antar pulau, negara, bahkan benua.

## d. Internet

Internet (*Interconnection Networking*) merupakan interkoneksi jaringan – jaringan komputer yang ada di seluruh dunia. “Koneksi antar jaringan komputer dapat dilakukan berkat dukungan protokol yang khas, yaitu *internet protocol*” (Sofana, 2008:5). Penjelasan mengenai perbandingan jarak dari jenis – jenis jaringan komputer dapat dilihat pada tabel 2.1.

**Tabel 2.1** Cakupan Area Jaringan Komputer (Sofana, 2008:5)

Jarak/cakupan (meter)	Contoh	Jenis
10 s/d 100	Ruangan	LAN
100 s/d 1000	Gedung	LAN
1000 s/d 10.000	Kampus	LAN
10.000 s/d 100.000	Kota	MAN
100.000 s/d 1.000.000	Negara	WAN
1.000.000 s/d 10.000.000	Benua	WAN
> 10.000.000	Planet	Internet

## 2.2 OSI dan TCP/IP

*Open System Interconnection* (OSI) dikembangkan oleh ISO (*International Organization for Standardization*) di Eropa pada tahun 1977. Model referensi ini

pada awalnya ditujukan untuk mengembangkan protokol – protokol jaringan, namun gagal diwujudkan. “Beberapa faktor penyebab gagalnya tujuan ini yaitu : OSI mirip dengan model referensi DARPA, OSI dianggap terlalu kompleks, OSI kurang diminati dibanding TCP/IP, dan adanya unsur politik pada pembuatan OSI” (Sofana,2008:80).

OSI merupakan sebuah model referensi yang digunakan sebagai panduan dalam mendesain protokol jaringan. OSI terdiri dari 7 *layer* yang masing – masing *layer* memiliki fungsi khusus. Pada model ini, paket data akan melalui suatu *layer* kemudian ke *layer* berikutnya. Penjelasan *layer – layer* pada model OSI dapat dilihat pada tabel 2.2.

**Tabel 2.2** Penjelasan Model OSI (CCNA, 2010)

Layer	Fungsi
(7) <i>Application</i>	<i>Layer</i> ini sebagai antarmuka antara aplikasi dan jaringan dimana paket ditransmisikan.
(6) <i>Presentation</i>	<ul style="list-style-type: none"> <li>- Melakukan <i>coding</i> dan konversi dari <i>layer application</i> untuk memastikan data dari sumbernya dapat dimengerti oleh aplikasi yang sesuai pada perangkat tujuan.</li> <li>- Kompresi data dengan cara yang dapat dimengerti oleh perangkat tujuan agar bisa didekompresi.</li> <li>- Enkripsi data pada saat proses transmisi, dan dekripsi data pada saat diterima oleh tujuan.</li> </ul>
(5) <i>Session</i>	<i>Layer</i> ini berfungsi mendefinisikan bagaimana suatu koneksi dibuat, dijaga, dan diakhiri.
(4) <i>Transport</i>	<ul style="list-style-type: none"> <li>- Pelacakan komunikasi individual antara aplikasi sumber dan tujuan.</li> <li>- Segmentasi data</li> <li>- Penyusunan segmen</li> <li>- Mengidentifikasi aplikasi yang berbeda</li> </ul>
(3) <i>Network</i>	Pengalamatan, enkapsulasi, <i>routing</i> , dan dekapsulasi.
(2) <i>Data Link</i>	<ul style="list-style-type: none"> <li>- Memungkinkan <i>layer</i> atas untuk mengakses media menggunakan teknik <i>framing</i>.</li> <li>- Melakukan kontrol pada data yang ditempatkan maupun data yang diterima dengan teknik <i>media access control</i> dan <i>error detection</i>.</li> </ul>
(1) <i>Physical</i>	Melakukan <i>encode</i> atau <i>decode</i> sinyal dari atau menuju tujuan.

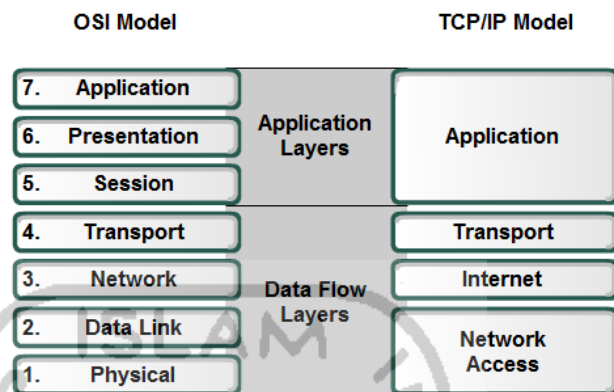
TCP/IP (*Transmission Control Protocol / Internet Protocol*) merupakan model protokol jaringan internet yang paling populer saat ini. Pada mulanya protokol ini diusulkan oleh DoD (*Department of Defense*) Amerika Serikat, dan dinamai model DARPA (*Defense Advanced Research Project Agency*) karena lembaga yang mengembangkannya adalah DARPA. Penggunaan model protokol ini dimulai pada jaringan ARPANET yang lahir pada tahun 1969. Seiring semakin berkembangnya jumlah *node* yang terhubung pada jaringan ARPANET, maka DARPA mendanai pembuatan protokol komunikasi yang kita kenal dengan nama TCP/IP.

TCP/IP menggunakan model dengan penerapan *layer*. Berbeda dengan OSI, TCP/IP hanya memiliki empat *layer*, yaitu *application*, *transport*, *internet*, dan *network access*. Penjelasan dari fungsi masing – masing *layer* pada TCP/IP dapat dilihat pada tabel 2.3.

**Tabel 2.3** Penjelasan Model TCP/IP (Sofana, 2008:89)

Layer	Fungsi
<i>Application</i>	Menampilkan data kepada <i>user</i> , melakukan <i>encoding</i> data juga kontrol dialog.
<i>Transport</i>	Mendukung proses komunikasi antar perangkat yang berbeda dan juga jaringan yang berbeda.
<i>Internet</i>	Berfungsi melakukan <i>routing</i> dan pembuatan paket IP menggunakan teknik enkapsulasi.
<i>Network Access</i>	Berfungsi meletakkan <i>frame-frame</i> data yang akan dikirim ke media jaringan, juga melakukan kontrol terhadap perangkat dan media yang digunakan dalam komunikasi data.

Perbandingan antara model referensi OSI dan model protokol TCP/IP dapat dilihat pada gambar 2.5.



Gambar 2.5 Perbandingan OSI dan TCP/IP (CCNA, 2010)

### 2.3 Routing Protocol

Di dalam jaringan internet, salah satu perangkat yang memegang peranan penting dalam proses pengiriman paket yaitu *router*. *Router* merupakan sebuah perangkat yang dapat mengirimkan paket data dari suatu jaringan ke jaringan yang lain. Ketika *router* yang satu dengan *router* yang lainnya akan mulai berkomunikasi, maka diperlukan penggunaan *routing protocol* yang dapat dimengerti oleh masing – masing *router*. Secara umum *routing protocol* dibagi menjadi dua, yaitu :

#### a. *Routing* statis

Pada *routing* statis, administrator memegang peranan penuh dalam mengelola informasi dari tabel *routing* setiap *router*. Administrator jaringan melakukan *entry* informasi *routing* ke dalam tabel *routing* secara manual. Kelebihan dari *routing* statis yaitu administrator memiliki “peta” penuh dari jaringan yang ia kelola. *Routing* statis cocok diterapkan pada *router* di jaringan yang hanya memiliki satu rute tujuan menuju jaringan yang lain.



b. *Routing* dinamis

Berbeda dengan *routing* statis, *routing* dinamis hanya memerlukan sedikit campur tangan manusia dalam mengelola tabel *routing*-nya. *Routing* dinamis sangat cocok diterapkan pada suatu jaringan yang memiliki lebih dari satu rute menuju jaringan lain. Berikut merupakan beberapa kelebihan dari *routing* dinamis (Sofana, 2008:148) :

- a. Tidak terlalu banyak membutuhkan campur tangan manusia sehingga mudah dalam dikelola.
- b. Dapat beradaptasi pada perubahan kondisi jaringan, seperti : putusnya suatu rute, penambahan rute, perubahan rute, dan lain lain.
- c. Rute ditentukan berdasarkan informasi yang dikirimkan oleh *router* lain.

Beberapa contoh *routing protocol* dinamis yaitu, RIP, RIPv2, EIGRP, OSPF, BGP, dan lain – lain.

### 2.3.1 OSPF (*Open Shortest Path First*)

*Link-state protocol* merupakan dasar dari *routing protocol* OSPF. Berbeda dengan *distance vector* (contoh: RIP, EIGRP, dan lain – lain), *link-state* mencari jalur terbaik dengan menggunakan algoritma Dijkstra yang berpatokan pada *cost* dari media penghubung antar *router*. *Cost* dari *link* dihitung dengan menggunakan rumus :

$$\text{Cost} = 10^8 / \text{bandwidth} \text{ (dalam bit/seconds) (CCNA, 2010)}$$

Semakin kecil *cost* menuju *router* tujuan, maka jalur tersebutlah yang akan dipilih oleh *link-state*. Nilai *cost* dari masing – masing jenis *interface* dapat dilihat pada tabel 2.4.

**Tabel 2.4** Penjelasan Cost Dari Beberapa Interface (CCNA, 2010)

Type Interface	Cost
Fast Ethernet	$10^8/100.000.000 = 1$
Ethernet	$10^8/10.000.000 = 10$
E1	$10^8/2.048.000 = 48$
T1	$10^8/1.544.000 = 64$
128 kbps	$10^8/128.000 = 781$
64 kbps	$10^8/64.000 = 1562$
56 kbps	$10^8/56.000 = 1785$

Setiap *router* akan membentuk LSP (*Link State Packet*) terlebih dahulu yang kemudian akan langsung di sebarakan ke *router* tetangga (*directly connected*). *Router* yang menerima LSP akan langsung menyebarkan paket tersebut ke *router* tetangga tanpa memprosesnya terlebih dahulu, sehingga mempercepat waktu konvergensi. Pada *link-state routing protocol*, *router* tidak mengirimkan *update* tabel *routing* secara berkala seperti *distance vector*. *Update* dikirimkan hanya pada saat pertama kali proses konvergensi *routing protocol*, ketika terjadi perubahan pada topologi, dan hanya diberikan pada *router* yang menganut SPF (*shortest path first*). Karakteristik dari *link-state* yaitu :

1. Memerlukan *memory*, CPU, dan *bandwidth* yang lebih besar dibandingkan dengan *distance vector* karena digunakan untuk membangun topologi utuh dari suatu *network*.
2. Menggunakan LSDB (*link state database*) untuk menjalankan algoritma SPF.

OSPF pertama kali dipublikasikan di RFC (*Request for Comments*) pada tahun 1989 dan disebut OSPFv1. OSPFv1 merupakan *routing protocol* yang sifatnya hanya berupa eksperimen dan tidak pernah disebarakan. Pada tahun 1991, OSPFv2 diperkenalkan oleh John Moy pada RFC 1247. OSPFv2 merupakan versi OSPF yang sering dipakai pada saat ini. Versi lebih lanjut dari *routing protocol* ini yaitu OSPFv3 yang telah mendukung penggunaan IPv6.

OSPF merupakan *routing protocol* yang cara kerjanya didasarkan pada *link-state protocol*. OSPF termasuk salah satu *routing protocol* IGP yang didesain

untuk dijalankan pada AS (*Autonomous System*) tunggal. “AS didefinisikan sebagai satu set *router* yang berada di bawah pengawasan administrasi tunggal” (Rafiudin, 2004:34). Salah satu kelebihan OSPF dibandingkan dengan RIP yaitu skalabilitas yang jauh lebih besar, dibandingkan dengan RIP yang hanya terbatas pada 15 *hop count* saja.

Sebelum dua *router* atau lebih dapat saling berkomunikasi terkait OSPF *neighbor*, mereka harus sepakat dalam tiga hal, yaitu *hello interval*, *dead interval*, dan tipe jaringan. *Hello interval* dari OSPF secara default yaitu 10 detik untuk multi akses dan *point to point*, 30 detik untuk *non-broadcast multiaccess* (NBMA) *segments* (contoh Frame Relay, ATM), sedangkan *dead interval*, yaitu waktu yang digunakan oleh *router* untuk menunggu *hello packet*, yaitu empat kali dari *hello interval*. jika *dead interval* sudah habis sebelum *router* mendapatkan *hello packet*, maka jalur tersebut otomatis dibuang dari LSDB, dan *router* akan langsung melakukan *flooding LSP*. *Hello interval* dan *dead interval* setiap *router* harus sama agar bisa saling terhubung.

untuk mengurangi trafik OSPF pada *multi access network*, OSPF akan memilih DR (*designated router*) dan BDR (*backup designated router*). DR bertanggung jawab untuk memberikan *update* ke semua *router* OSPF, sedangkan BDR akan menggantikan tugas DR bila DR yang ada *down*.

### 2.3.2 BGP (*Border Gateway Protocol*)

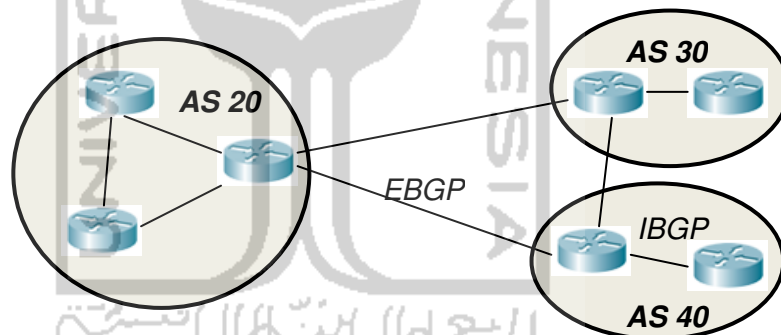
BGP merupakan sebuah *routing protocol inter-Autonomous System*. *Routing protocol* BGP didasarkan pada *path vector protocol*. “*Path vector protocol* tidak bergantung pada *cost* untuk mencapai tujuan tertentu, melainkan mengandalkan analisis dari jalur untuk mencapai tujuan yang bebas dari *loop*” (White, et. al., 2004).

Protokol ini dijelaskan antara lain dalam RFC 4271, 4276, dan 4277. “Fungsi utama BGP yaitu bertukar informasi *network* yang *reacheable* oleh

sistem BGP lain” (Rafiudin, 2004:21). BGP adalah salah satu *routing protocol* EGP yang menggunakan TCP port 179 untuk membangun dan menjaga koneksi antar *router*. Perbedaan mencolok antara EGP dan IGP yaitu : (White, et. al., 2004).

1. IGP secara umum fokus pada kecepatan waktu konvergensi dan mengasumsikan keseragaman kebijakan pada satu AS.
2. EGP fokus kepada penyebaran *policy* dan menjaga stabilitas.

Ketika BGP digunakan antar AS yang berbeda maka disebut EBG (External BGP), sedangkan BGP yang digunakan untuk bertukar rute dalam sebuah AS yang sama disebut IBGP (Internal BGP). Contoh topologi sederhana dari pengimplementasian BGP dapat dilihat pada gambar 2.6.



**Gambar 2.6** Contoh Topologi Sederhana BGP

BGP merupakan *routing protocol* yang sangat *powerful* dan *scalable*, hal ini dapat dilihat melalui fakta bahwa BGP adalah *routing protocol* yang digunakan oleh internet saat ini. “BGP hanya mendukung *policy* yang mengikuti paradigma *hop-by-hop* dan karena internet menggunakan paradigma ini maka BGP menjadi sangat mungkin untuk diaplikasikan sebagai *routing protocol* AS diantara koneksi internet” (Rafiudin, 2004:22). Yang dimaksud *hop* pada BGP berbeda dengan *hop* pada *distance vector*. Satu *hop* pada BGP dihitung ketika paket melewati satu AS bukan satu *router*.

Untuk mengurangi ukuran tabel *routing*, BGP juga menerapkan CIDR (*Classless Interdomain Routing*). Contoh kasus penerapan CIDR yaitu, diasumsikan bahwa ISP mempunyai blok ip address kelas C, kemudian masing – masing ip address diberikan kepada pelanggannya. Tanpa CIDR, ISP akan meng*advertise* 256 alamat yang ada pada blok IP tersebut ke semua *peer* BGP. Namun dengan CIDR, BGP dapat meng*summarize* alamat dan meng*advertise* hanya satu blok alamat saja.

BGP *neighbor* saling bertukar informasi *routing* saat koneksi TCP pertama kali dilakukan. BGP tidak mengirimkan *update routing* secara berkala, *update* hanya dikirimkan jika terjadi perubahan pada tabel *routing* dan yang dikirimkan hanya bagian yang berubah saja.

Sistem BGP memiliki empat jenis pesan, yaitu : *open*, *update*, *keepalive*, dan *notification*. Penjelasan mengenai empat jenis pesan BGP dapat dilihat pada tabel 2.5.

**Tabel 2.5** BGP Message (Rafiudin, 2004:36)

Message	Keterangan
<i>Open</i>	Setelah koneksi TCP terjalin antar <i>router</i> BGP, maka <i>router</i> tersebut akan saling bertukar pesan <i>open</i> . Setelah itu maka antar <i>router</i> tadi sudah dapat bertukar data.
<i>Update</i>	<i>Router</i> BGP mengirim pesan <i>update</i> untuk melakukan pertukaran informasi mengenai <i>reachabilitas</i> jaringan.
<i>Keepalive</i>	Sistem BGP akan bertukar pesan <i>keepalive</i> untuk mengetahui apakah sebuah jalur dianggap <i>fail</i> atau tidak eksis lagi.
<i>Notification</i>	Pesan ini akan dikirim jika terdeteksi adanya <i>error</i> . Begitu pesan terkirim sesi BGP dan koneksi TCP antar sistem – sistem BGP akan ditutup.

BGP memiliki beberapa atribut yang digunakan untuk menentukan rute terbaik ketika tersedia beberapa jalur menuju tujuan. Tidak semua atribut kompatibel dengan semua *router*, beberapa atribut hanya dimiliki oleh *router* dengan pabrikan tertentu. Atribut BGP dibagi menjadi beberapa bagian yaitu (White, et. al., 2004) :

1. Atribut *well-known mandatory*

Atribut ini didukung oleh semua perangkat *router* dan pasti ada dalam pesan *update*.

2. Atribut *well-known discretionary*

Atribut ini didukung oleh semua perangkat *router* dan bisa ada pada pesan *update* bisa juga tidak, tergantung keperluan.

3. Atribut *optional transitive*

Atribut ini tidak didukung oleh semua *router* dan harus dijaga dengan melakukan *advertise* ke semua *peer* BGP.

4. Atribut *optional nontransitive*

Atribut ini tidak didukung oleh semua *router*. Jika sebuah *update* berisi atribut ini, maka *update* ini harus diiklankan ke BGP *peer* lainnya tanpa mengiklankan atribut yang tidak dikenali oleh *router* lainnya.

Salah satu atribut *well-known* terkait dengan penelitian pada tugas akhir ini. Penjelasan atribut *well-known* BGP dapat dilihat pada tabel 2.6.

**Tabel 2.6** Atribut Well-Known BGP (White, et. al., 2004)

Atribut	Keterangan
AS Path	Atribut ini bersifat <i>well-known mandatory</i> . Atribut ini berisi daftar semua jalur AS.
Next hop	Atribut ini bersifat <i>well-known mandatory</i> . Atribut ini berperan dalam mendefinisikan ip <i>address border router</i> yang akan digunakan sebagai <i>next-hop</i> ke AS tujuan.
Origin Code	Atribut ini bersifat <i>well-known mandatory</i> . Atribut ini dibuat oleh AS yang memulai informasi <i>routing</i> . Ada 3 tipe Origin yaitu : IGP, EGP, Incomplete. Jalur terbaik yang akan dipilih yaitu Origin dengan tipe paling rendah. Urutan tipe Origin mulai dari yang terendah : IGP lebih rendah dibandingkan EGP, dan EGP lebih rendah dibandingkan Incomplete.
Local Preference	Atribut ini bersifat <i>well-known discretionary</i> . Atribut ini merepresentasikan derajat preferensi dari suatu jalur. Semakin tinggi nilai Local Preference maka jalur tersebut akan dipilih.
Atomic Aggregate	Atribut ini bersifat <i>well-known discretionary</i> .

Berikut merupakan penjelasan bagaimana BGP menentukan rute terbaiknya baik itu menggunakan parameter atribut BGP maupun parameter lainnya (Cisco, 2012) :

1. Jalur dengan nilai Weight tertinggi akan dipilih. Weight merupakan *proprietary* dari cisco.
2. Jalur dengan nilai Local Preference tertinggi akan dipilih.
3. Lebih memilih jalur yang secara lokal berasal melalui jaringan atau perintah agregat BGP atau melalui redistribusi dari IGP.
4. Jalur dengan AS Path terpendek akan dipilih.
5. Jalur dengan tipe Origin terendah akan dipilih.
6. Jalur dengan multi-exit discriminator (MED) terendah akan dipilih
7. Lebih memilih EBGP dibandingkan IBGP.
8. Lebih memilih jalur metrik IGP terendah menuju BGP *next hop*.
9. Menentukan apakah beberapa jalur memerlukan instalasi pada tabel *routing* untuk BGP *multipath*.
10. Jika ada beberapa jalur eksternal, maka akan dipilih jalur yang diterima pertama kali.
11. Lebih memilih rute yang datang dari *router* BGP dengan *router ID* terendah.
12. Jika originator atau *router ID* sama pada beberapa jalur, maka yang akan dipilih yaitu jalur dengan panjang cluster minimum.
13. Lebih memilih jalur yang datang dari alamat tetangga BGP terendah.

#### 2.4 Simulator OPNET Modeler 14.0

Opnet Modeler 14.0 merupakan sebuah perangkat lunak yang digunakan untuk melakukan simulasi, baik itu terhadap lingkungan, perangkat, protokol, aplikasi, dan desain komunikasi pada infrastruktur teknologi informasi. Simulasi yaitu mengacu pada pengertian suatu proses untuk menggambarkan semirip mungkin suatu hal berikut lingkungan sekitarnya sehingga didapatkan hasil yang

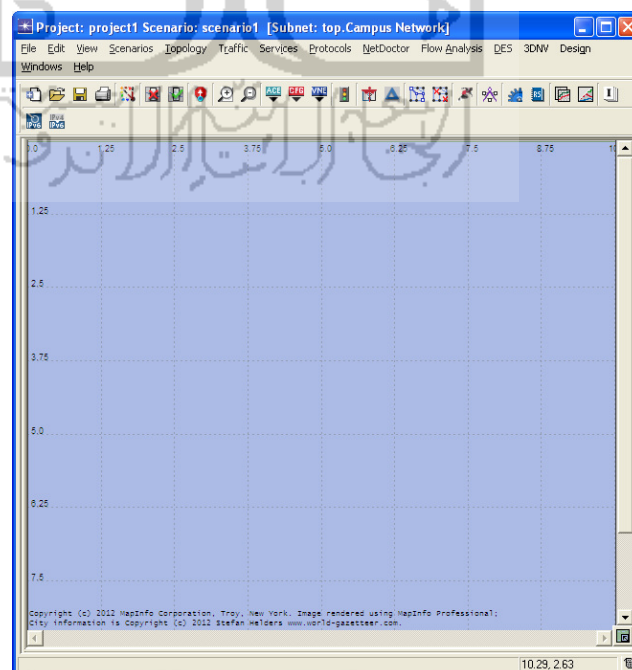
dapat merepresentasikan ciri fisik, tampilan dan karakteristiknya. Simulator OPNET digunakan untuk mendapatkan statistik yang dihitung secara sistematis dari suatu model infrastruktur IT dengan cakupan wilayah mulai dari skala kampus, kantor, bahkan dunia. Alur kerja OPNET dapat dilihat pada gambar 2.7.



**Gambar 2.7** Workflow OPNET Modeler (Svensson, 2003)

#### 2.4.1 *Project Editor*

Hal paling utama yang digunakan untuk membangun simulasi di OPNET yaitu *project editor*. *Project editor* digunakan untuk membuat skema *project*, topologi, juga skenario yang mendukung *project*. *Project editor* terdiri dari area kerja dan *tools* yang dapat digunakan untuk membangun sebuah skenario *project*. Tampilan awal dari *project editor* dapat dilihat pada gambar 2.8.

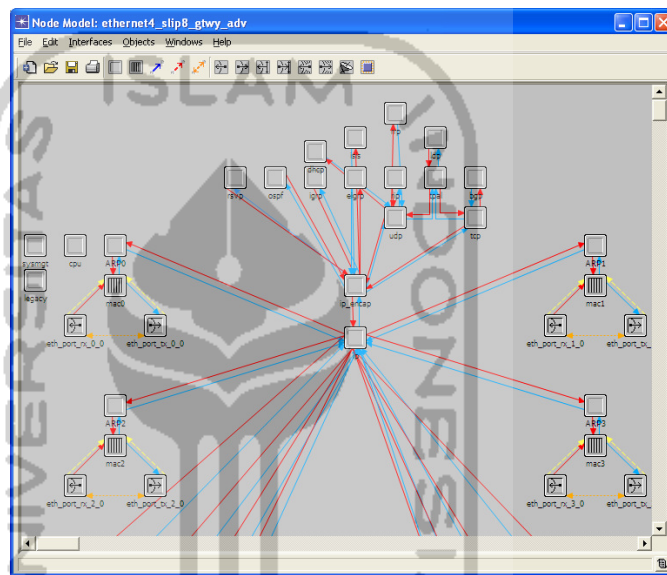


**Gambar 2.8** Project Editor Pada OPNET



### 2.4.2 Node Editor

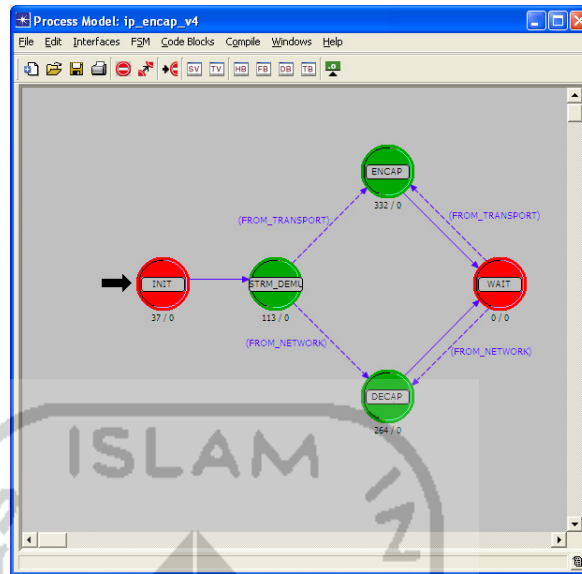
*Node editor* digunakan untuk membuat *node* model. *Node* model memiliki struktur modular. Misalnya saja *node* yang berupa *router* memiliki beberapa komponen penyusun didalamnya seperti, *protocol*, CPU, *system management*, *ip encapsulation*, dan lain lain. Tampilan dari *node editor* dapat dilihat pada gambar 2.9.



Gambar 2.9 Node Model

### 2.4.3 Process Model Editor

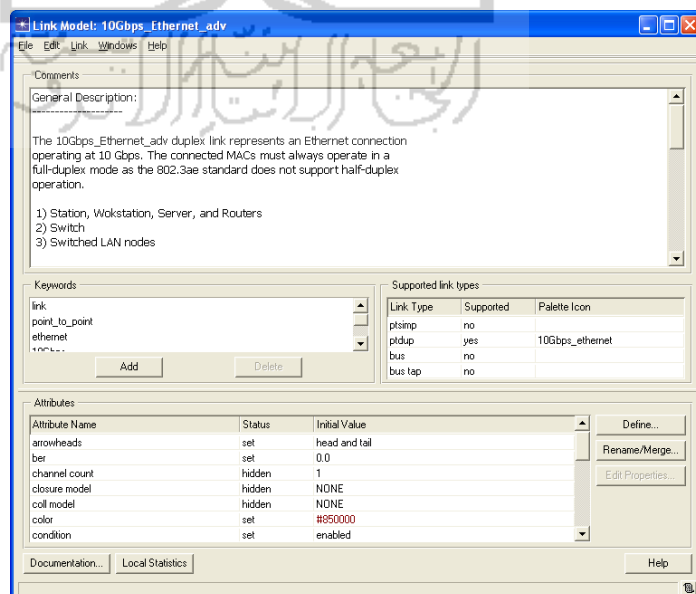
*Process* model mendasari fungsionalitas dari sebuah *node* model. Pada *process* model didefinisikan bagaimana alur kerja dari *node* model. Setiap operasi pada *process* model terbentuk melalui bahasa C atau C++. Tampilan dari *process model editor* dapat dilihat pada gambar 2.10.



**Gambar 2.10** Process Model

#### 2.4.4 Link Model Editor

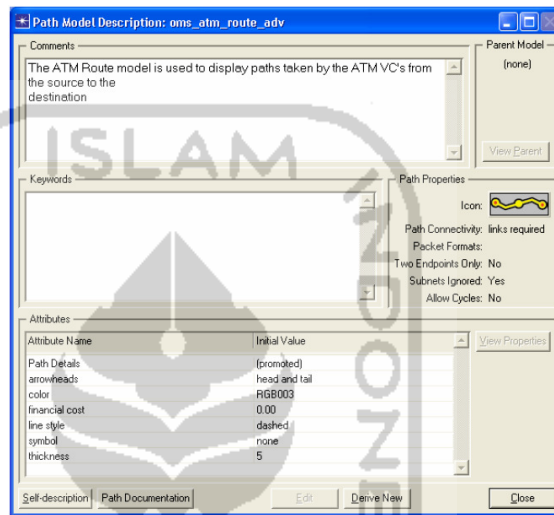
*Link* model mendeskripsikan detail dari suatu media penghubung antar *node*. Dengan menggunakan *link model editor*, memungkinkan pengguna untuk membuat jenis *link* baru. Tampilan dari *link model editor* dapat dilihat pada gambar 2.11.



**Gambar 2.11** Link Model Editor

### 2.4.5 Path Editor

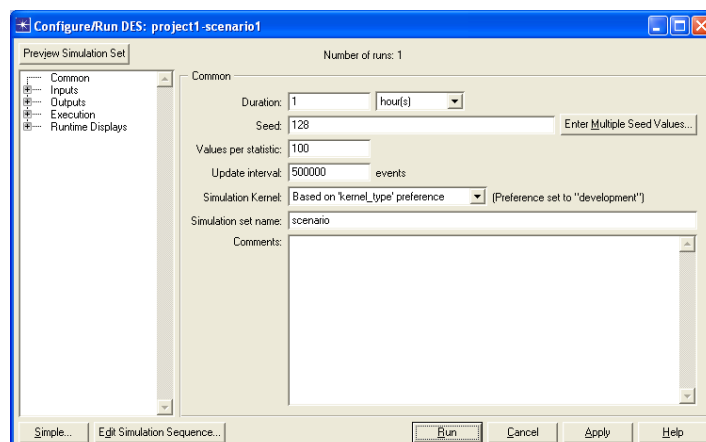
Digunakan untuk membuat objek *path* baru yang mendefinisikan sebuah trafik rute. “Setiap model protokol yang menggunakan koneksi logis seperti MPLS, ATM, Frame Relay, dapat menggunakan *path* untuk trafik rutanya” (Svensson, 2003). Tampilan dari *path editor* dapat dilihat pada gambar 2.12.



Gambar 2.12 Path Editor (Svensson, 2003)

### 2.4.6 Simulation Sequence Editor

Editor ini digunakan untuk melakukan konfigurasi terhadap jalannya simulasi dari suatu *project*. Tampilan dari *editor* simulasi OPNET dapat dilihat pada gambar 2.13.



Gambar 2.13 Editor Simulasi OPNET

## BAB III

### METODOLOGI PENELITIAN

#### 3.1 Analisis Kebutuhan Sistem

Pada bab ini akan dibahas mengenai kebutuhan sistem yang digunakan dalam penelitian mengenai analisis kinerja *routing protocol* BGP menggunakan BGP *policy*.

##### 3.1.1 Perangkat Keras yang Dibutuhkan

Kebutuhan perangkat keras minimum yang dibutuhkan untuk menjalankan OPNET yaitu :

- a. CPU
  - Minimum: 2,0 GHz untuk Windows, 1,0 GHz untuk Linux menggunakan x86, EM64T, AMD x86, atau AMD64.
  - Rekomendasi: 3.0 + GHz menggunakan x86, EM64T, AMD x86, atau AMD64 (dual-core).
- b. RAM
  - Minimum: 512 MB
  - Rekomendasi: 1-2 GB
- c. *System File Space*

3 GB dengan tambahan sampai 2 GB *space* kosong yang digunakan selama proses instalasi.
- d. *Working File Space*

Lebih dari 100 MB untuk *temporary* dan *log files*.
- e. *Display*

Resolusi layar minimal yaitu 1024x768.

##### 3.1.2 Perangkat Lunak yang Dibutuhkan

Kebutuhan minimum perangkat lunak yang digunakan dalam penelitian ini adalah sebagai berikut:

a. Sistem Operasi

- Microsoft

Windows 7 Professional (32 bit dan 64 bit), Windows Vista Business (32 bit dan 64 bit), Windows XP Professional (32 bit dan 64 bit), Windows Server 2008 (32 bit dan 64 bit), Windows Server 2003 (32 bit dan 64 bit) dan Windows Server 2003 R2 (32 bit dan 64 bit).

- Red Hat

Red Hat Enterprise Linux 5 (v2.6.18 Linux kernel), Red Hat Enterprise Linux 4 (v2.6 Linux kernel).

- Fedora

Fedora Linux 6 (v2.6.18 Linux kernel).

b. Perangkat Lunak Pendukung

- Untuk linux dibutuhkan gcc v3.4 atau lebih, sedangkan untuk windows dibutuhkan : Microsoft Visual Studio .NET 2010, Microsoft Visual C++ 2010 Express Edition, Microsoft Visual Studio.NET 2008, Microsoft Visual C++ 2008 Express Edition, Microsoft Visual Studio .NET 2005, Microsoft Visual Studio .NET 2003 (Opnet, 2011).
- OPNET Modeler 14.0 versi edukasi
- VMware

### 3.2 Instalasi dan Konfigurasi Sistem

Berikut merupakan beberapa tahapan yang digunakan penulis dalam melakukan instalasi OPNET Modeler 14.0 versi edukasi :

a. Instalasi Sistem Operasi

Sistem Operasi yang digunakan yaitu Windows XP *service pack 2* yang dijalankan menggunakan *virtual machine* VMware.

b. Instalasi Perangkat Lunak Pendukung

Sebelum OPNET Modeler 14.0 versi edukasi diinstal, sebelumnya penulis melakukan instalasi perangkat lunak pendukung untuk OPNET yaitu Microsoft Visual Studio 2005.

c. Instalasi OPNET Modeler 14.0

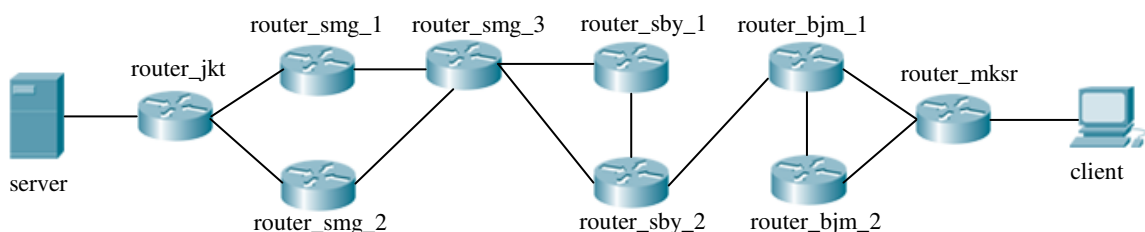
Setelah semua kebutuhan minimum OPNET terinstal dengan benar, tahap selanjutnya yaitu melakukan instalasi OPNET Modeler 14.0 berikut lisensi versi edukasi.

d. Konfigurasi *Environment* pada Sistem Operasi

Pada tahapan ini, penulis melakukan konfigurasi dengan menambahkan *environment variable* pada Windows XP. Kemudian penulis memastikan OPNET sudah berjalan dengan benar.

### 3.3 Desain Sistem

Topologi yang akan dibuat pada OPNET menggunakan peta Indonesia tepatnya di kota Jakarta, Semarang, Surabaya, Banjarmasin, dan Makassar. Beberapa objek yang digunakan pada penelitian ini yaitu yaitu 9 buah *router*, 1 buah *client*, dan 1 buah *server* dengan *link* untuk koneksi antar *router* menggunakan kabel serial tipe DS3 dengan data *rate* 44,736 Mbps dan fast ethernet dengan data *rate* 100 Mbps untuk *link* dari LAN ke *router* dan juga dari *server* ke *router*. Desain sistem dapat dilihat pada gambar 3.1.



**Gambar 3.1** Desain Sistem

Skenario yang akan dibuat sebanyak 12 buah yaitu OSPF\_Simple, BGP\_Simple, BGP\_Policy, OSPF\_Http, BGP\_Http, BGP\_Policy\_Http, OSPF\_Voice, BGP\_Voice, BGP\_Policy\_Voice, OSPF\_Video, BGP\_Video, dan BGP\_Policy\_Video. Pemilihan skenario di sini hanya dibatasi pada beberapa layanan di internet yang penulis anggap lebih dominan penggunaannya pada saat ini, yaitu website, VOIP, dan *video conference*.

### 3.3.1 Skenario OSPF\_Simple

Pada skenario ini diimplementasikan *routing protocol* OSPF pada setiap *router*. Semua *router* di topologi ini dianggap pada satu AS yang sama, dikarenakan OSPF bukanlah *routing protocol* yang ditujukan untuk keperluan EGP. Tidak ada perubahan pada atribut yang menentukan rute terbaik pada skenario ini, semua parameter dianggap pada kondisi *default*. Tujuan utama dari skenario ini yaitu mengamati trafik OSPF, dan penggunaan *resource* CPU.

### 3.3.2 Skenario BGP\_Simple

Skenario ini merupakan pengembangan dari OSPF\_Simple. Setiap *router* dibedakan AS nya berdasarkan wilayah. Wilayah Jakarta AS *number*nya yaitu 2500, wilayah Semarang AS *number*nya yaitu 3500, wilayah Surabaya AS *number*nya yaitu 4500, wilayah Banjarmasin AS *number*nya yaitu 5500, dan wilayah Makassar AS *number*nya yaitu 6500. Pada setiap *router* yang bersinggungan langsung dengan AS lain akan diimplementasikan *routing protocol* BGP secara *default*. Tujuan utama dari skenario ini yaitu mengamati trafik BGP, dan penggunaan *resource* CPU.

### 3.3.3 Skenario BGP\_Policy

Skenario ini merupakan versi lebih lanjut dari skenario BGP\_Simple. Dimana seluruh konfigurasi merupakan hasil duplikat dari skenario BGP\_Simple. Yang membedakan yaitu diterapkan nya sebuah *policy* yang berkaitan dengan salah satu atribut yang menentukan rute terbaik menuju tujuan. Atribut yang

dimaksud adalah Local Preference. Tujuan utama dari skenario ini yaitu mengamati trafik BGP, dan penggunaan *resource* CPU.

#### **3.3.4 Skenario OSPF\_Http, BGP\_Http, dan BGP\_Policy\_Http**

Konsep topologi dan parameter dari ketiga skenario ini sama dengan pada skenario sebelumnya, OSPF\_Simple, BGP\_Simple, dan BGP\_Policy. Bedanya yaitu ditambahkan aplikasi web *service*, yang akan *direquest* oleh klien. Tujuan utama dari skenario ini yaitu mengamati perbandingan dari ketiga skenario terhadap waktu respon halaman web, dan trafik yang diterima oleh klien.

#### **3.3.5 Skenario OSPF\_Voice, BGP\_Voice, dan BGP\_Policy\_Voice**

Konsep topologi dan parameter dari ketiga skenario ini sama dengan pada skenario sebelumnya, OSPF\_Simple, BGP\_Simple, dan BGP\_Policy. Bedanya yaitu ditambahkan aplikasi VOIP, yang akan *direquest* oleh klien. Tujuan utama dari skenario ini yaitu mengamati perbandingan dari ketiga skenario terhadap variasi *delay* yang ditimbulkan dari VOIP, dan trafik yang diterima oleh klien.

#### **3.3.6 Skenario OSPF\_Video, BGP\_Video, dan BGP\_Policy\_Video**

Konsep topologi dan parameter dari ketiga skenario ini sama dengan pada skenario sebelumnya, OSPF\_Simple, BGP\_Simple, dan BGP\_Policy. Bedanya yaitu ditambahkan aplikasi *video conference*, yang akan *direquest* oleh klien. Tujuan utama dari skenario ini yaitu mengamati perbandingan dari ketiga skenario terhadap variasi *delay* yang ditimbulkan dari *video conference*, dan trafik yang diterima oleh klien.



### 3.4 Konfigurasi Simulasi

Hal pertama yang dilakukan yaitu membuat sebuah project baru pada OPNET. Berikut dijelaskan langkah – langkah yang penulis lakukan :

#### 3.4.1 Skenario OSPF\_Http

1. Membuat file project baru.

Pilih **File** → **New...** → **Project** → **Ok**

**Project Name** = Kinerja\_BGP

**Scenario Name** = OSPF

Selanjutnya **Create empty scenario** → **Next**

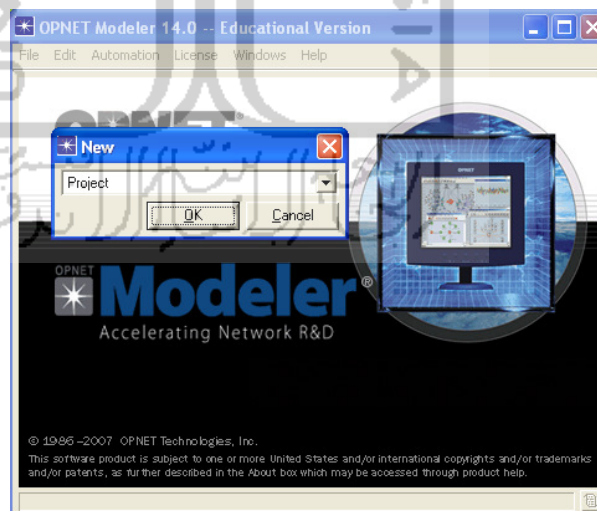
**Network Scale** = **World** → **Next**

**Selected (background first)** → **Next**


Model perangkat didefinisikan nanti, **Next**

**Review** → **Finish**

Contoh tampilan dari pembuatan *project* baru dapat dilihat pada gambar 3.2.



**Gambar 3.2** Pembuatan Project Baru

2. Selanjutnya yaitu memilih perangkat yang akan digunakan untuk simulasi, pada menu **Object Palette** (ikon ). Dipilih beberapa perangkat berikut :

- a. 9 buah **ethernet4\_slip8\_gtwy** untuk *router*.
- b. 1 buah **ethernet\_server** untuk *server*.
- c. 1 buah **100BaseT\_LAN** untuk *client*.

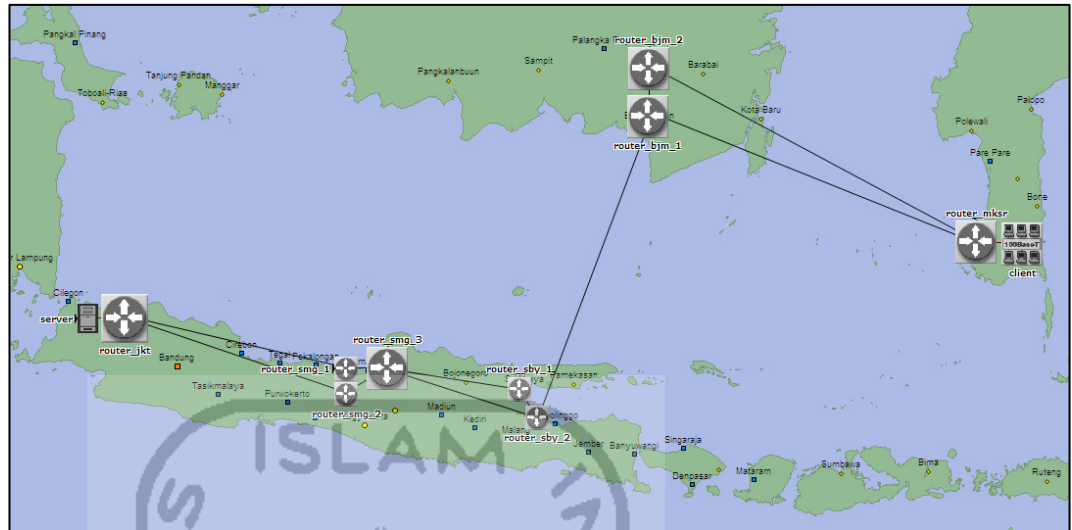
*Workstation* yang digunakan berjumlah 30 buah, jadi dilakukan modifikasi pada parameter di *client*. Klik kanan pada **client**, **Edit Attributes** → **LAN** → **Number of Workstations = 30**.

*Link* yang digunakan untuk koneksi antar *router* yaitu **PPP\_DS3** (kabel serial tipe DS3 dengan data *rate* 44,736 Mbps) dan **100BaseT** (kabel fast ethernet dengan data *rate* 100 Mbps) untuk *link* dari *router* ke *client* dan dari *server* ke *router*. Kemudian masing – masing *node* diberi nama seperti pada tabel 3.1.

**Tabel 3.1** Penjelasan Perangkat Pada Desain Sistem

Kota	Perangkat
Jakarta	a. server b. router_jkt
Semarang	a. router_smg_1 b. router_smg_2 c. router_smg_3
Surabaya	a. router_sby_1 b. router_sby_2
Banjarmasin	a. router_bjm_1 b. router_bjm_2
Makassar	a. router_mksr b. client

Caranya klik kanan pada *node* yang dimaksud kemudian pilih **Set Name**. Topologi awal dari *project* ini akan tampak seperti gambar 3.3.



**Gambar 3.3** Topologi Project Kinerja\_BGP

3. Selanjutnya beralih ke konfigurasi pada *router*. *Router* diseleksi semuanya menggunakan shift+klik kiri. Setelah itu klik kanan dan pilih **Edit Attributes**. Kemudian centang pada opsi **Apply to selected objects**. Berikut beberapa parameter pada *router* yang dimodifikasi :

- a. *Routing Protocol* OSPF

**IP Routing Protocols → OSPF Parameters → Processes → 1**

**Address Family = IPv4 – Multicast**

Konfigurasi OSPF dapat dilihat pada gambar 3.4.

OSPF Parameters	(...)
Processes	(...)
Number of Rows	1
1	
Process Tag	1
VRF Name	None
Process Parameters	Default
Version	v2
Address Family	IPv4 - Multicast
Instance ID	Not Configured

**Gambar 3.4** Konfigurasi Address Family pada OSPF Parameters

Kemudian pada **Process Parameters**, **Redistribution → Routing Protocols → Directly Connected → Redistribute w/ Default**.

Konfigurasi ini bertujuan agar setiap *link* yang terkoneksi ke *router* mendistribusikan *routing protocol* OSPF. Konfigurasi redistribusi OSPF dapat dilihat pada gambar 3.5.

Redistribution	(...)
Default Metric	Not Configured
Routing Protocols	(...)
Directly Connected	Redistribute w/ Default
Static	No Redistribution
BGP	No Redistribution
EIGRP	No Redistribution
IGRP	No Redistribution
IS-IS	No Redistribution
OSPF	No Redistribution
RIP	No Redistribution
RIPng	No Redistribution

**Gambar 3.5** Konfigurasi Redistribusi Pada OSPF

b. *Reports*

**Reports → OSPF Routing Table = Export at End of Simulation**

Konfigurasi untuk menampilkan tabel *routing* OSPF dapat dilihat pada gambar 3.6.

Reports	
BGP Routing Table	Do Not Export
EIGRP Routing Table	Do Not Export
IGRP Routing Table	Do Not Export
IP Forwarding Table	Do Not Export
IP Multicast Group-to-RP Table	Do Not Export
IS-IS Routing Table	Do Not Export
OSPF Link State Database	Do Not Export
OSPF Routing Table	Export at End of Simulation
PIM-SM Routing Table	Do Not Export
RIP Routing Table	Do Not Export
RIPng Routing Table	Do Not Export
VRF Table	Do Not Export

**Gambar 3.6** Report Pada Tabel Routing OSPF

c. Aktivasi *interface* loopback pada setiap *router*


**IP → IP Routing Parameters → Loopback Interfaces →**

**Number of Rows = 1**

Konfigurasi untuk mengaktifkan *interface* loopback dapat dilihat pada gambar 3.7.

IP Routing Parameters	(...)
Router ID	Auto Assigned
Autonomous System Number	Auto Assigned
Interface Information (12 Rows)	(...)
Aggregate Interfaces	None
Loopback Interfaces	(...)
Number of Rows	1
LBO	...

**Gambar 3.7** Aktivasi Interface Loopback

4. Kemudian jalankan simulasi pertama kali untuk mendapatkan *ip address* pada tiap perangkat. Jalankan simulasi dengan klik pada tombol , durasi simulasi diatur menjadi **10 menit**. Pada tab hierarki **Inputs** → **Global Attributes**, nilai atribut diubah menjadi sebagai berikut :

- a. **IP** → **IP Interface Addressing Mode = Auto Addressed/Export**

Konfigurasi ini bertujuan agar tiap *interface* yang aktif mendapat konfigurasi alamat secara otomatis dari OPNET dan juga alamat tadi akan di*export* dalam format *file* yang dapat dibaca OPNET agar mudah dalam melihat *interface* dan juga alamatnya.

- b. **IP** → **IP Version Preference = IPv4**

- c. **IP** → **IP Routing Table Export/Import = Export**

Konfigurasi ini dilakukan agar tabel *routing* dari tiap *router* di*export* dan dapat dilihat.

- d. **Simulation Efficiency** → **BGP Sim Efficiency Mode** dan **OSPF Sim Efficiency = Disabled**

Konfigurasi ini ditujukan agar BGP pada skenario selanjutnya dan OSPF terus melakukan *update* tabel *routing*.

Setelah itu klik **Apply** kemudian **Run** untuk menjalankan simulasi.

Konfigurasi atribut global simulasi dapat dilihat pada gambar 3.8.

IP	
IP Dynamic Routing Protocol	Default
IP Interface Addressing Mode	Auto Addressed/Export
IP Routing Table Export/Import	Export
IP Routing Table Source	Flow Analysis
IP Version Preference	IPv4
IPv6 Configuration	Consider
IPv6 Interface Address Export	Disabled
Interface Buffer Congestion Threshold	0.8
Routing Activity Idle Timer (seconds)	20
MANET	
MPLS	
Simulation Efficiency	
ARP Sim Efficiency	Enabled
BGP Sim Efficiency Mode	Disabled
EIGRP Sim Efficiency	Enabled
EIGRP Stop Time (seconds)	365
IGMP Sim Efficiency	Enabled
IGRP Sim Efficiency	Enabled
IGRP Stop Time (seconds)	365
IPv6 ND Simulation Efficiency	Disabled
ISIS Sim Efficiency	Enabled
ISIS Stop Time (seconds)	260
LACP Simulation Efficiency	Enabled
OSPF Sim Efficiency	Disabled

**Gambar 3.8** Konfigurasi Simulasi

5. Untuk mendapatkan tabel ip *address*, perlu dilakukan *refresh* pada direktori *project* agar hasil *export ip address* terlihat.

**File → Manage Model Files → Refresh Model Directories**

Untuk menampilkan informasi ip *address* dan juga *interface* yang digunakan yaitu dengan cara **File → Open →** pada tipe *file* ganti dengan **Generic Data File**, kemudian pilih *file* **Kinerja\_BGP-OSPF-DES-1-ip\_addresses.gdf**. Daftar ip *address* pada *project* ini dapat dilihat pada tabel 3.2.

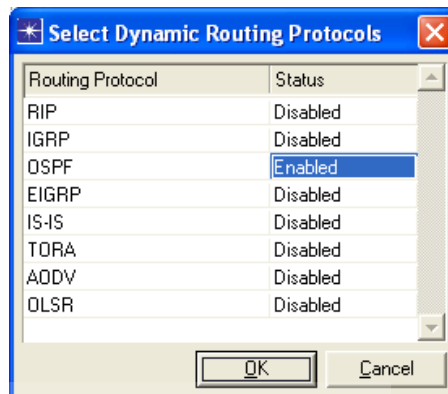
**Tabel 3.2** Informasi IP Address dan Link

Nama Perangkat	Iface	Ip Address	Link
server	IF0	192.0.0.1/24	server <-> router_jkt
client	IF0	192.0.20.2/24	router_mkxr <-> client
router_jkt	IF0	192.0.0.2/24	server <-> router_jkt_1
	IF10	192.0.1.1/24	router_jkt <-> router_smg_1
	IF11	192.0.2.1/24	router_jkt <-> router_smg_2
	LB0	192.0.3.1/24	Not connected to any link
router_smg_1	IF10	192.0.1.2/24	router_jkt <-> router_smg_1
	IF11	192.0.4.1/24	router_smg_1 <-> router_smg_3
	LB0	192.0.5.1/24	Not connected to any link

router_smg_2	IF10 IF11 LB0	192.0.2.2/24 192.0.6.1/24 192.0.7.1/24	router_jkt <-> router_smg_2 router_smg_2 <-> router_smg_3 Not connected to any link
router_smg_3	IF4 IF5 IF10 IF11 LB0	192.0.8.1/24 192.0.9.1/24 192.0.4.2/24 192.0.6.2/24 192.0.10.1/24	router_smg_3 <-> router_sby_1 router_smg_3 <-> router_sby_2 router_smg_1 <-> router_smg_3 router_smg_2 <-> router_smg_3 Not connected to any link
router_sby_1	IF10 IF11 LB0	192.0.8.2/24 192.0.11.1/24 192.0.12.1/24	router_smg_3 <-> router_sby_1 router_sby_1 <-> router_sby_2 Not connected to any link
router_sby_2	IF4 IF10 IF11 LB0	192.0.13.1/24 192.0.9.2/24 192.0.11.2/24 192.0.14.1/24	router_sby_2 <-> router_bjm_1 router_smg_3 <-> router_sby_2 router_sby_1 <-> router_sby_2 Not connected to any link
router_bjm_1	IF4 IF10 IF11 LB0	192.0.15.1/24 192.0.13.2/24 192.0.16.1/24 192.0.17.1/24	router_bjm_1 <-> router_mkrs router_sby_2 <-> router_bjm_1 router_bjm_1 <-> router_bjm_2 Not connected to any link
router_bjm_2	IF10 IF11 LB0	192.0.16.2/24 192.0.18.1/24 192.0.19.1/24	router_bjm_1 <-> router_bjm_2 router_bjm_2 <-> router_mkrs Not connected to any link
router_mkrs	IF0 IF10 IF11 LB0	192.0.20.1/24 192.0.15.2/24 192.0.18.2/24 192.0.21.1/24	router_mkrs <-> client router_bjm_1 <-> router_mkrs router_bjm_2 <-> router_mkrs Not connected to any link

#### 6. Aktivasi OSPF Pada *Interface*

Untuk mengaktifkan OSPF pada *interface* yang bersinggungan langsung dengan *router* lain caranya yaitu, klik kanan pada *router*, **Edit Attributes** → **IP** → **IP Routing Parameters** → **Interface Information** → pilih *interfacenya*, pada **Routing Protocols** pilih opsi **Enabled** hanya pada **OSPF**. Untuk *interface* yang tidak digunakan **Disabled** semua opsi. Konfigurasi aktivasi OSPF pada *interface* dapat dilihat pada gambar 3.9.



**Gambar 3.9** Aktivasi OSPF Pada Interface

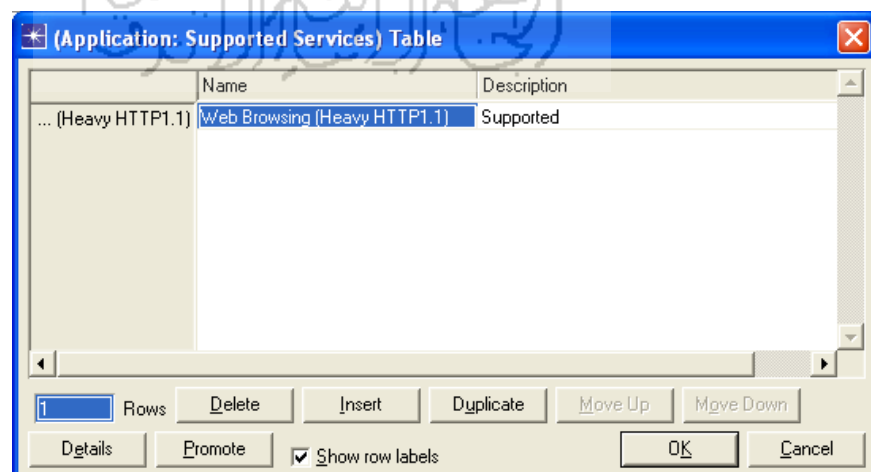
#### 7. Konfigurasi Layanan Pada *Server*

Pada skenario OSPF\_Http, layanan yang akan didistribusikan oleh *server* yaitu *web service*. Untuk mengaktifkan layanan ini pada *server*, klik kanan pada *server*, kemudian **Edit Attributes**.

**Application: Supported Services** → **Edit Name = Web Browsing (Heavy Http 1.1)**

**Description = Supported**

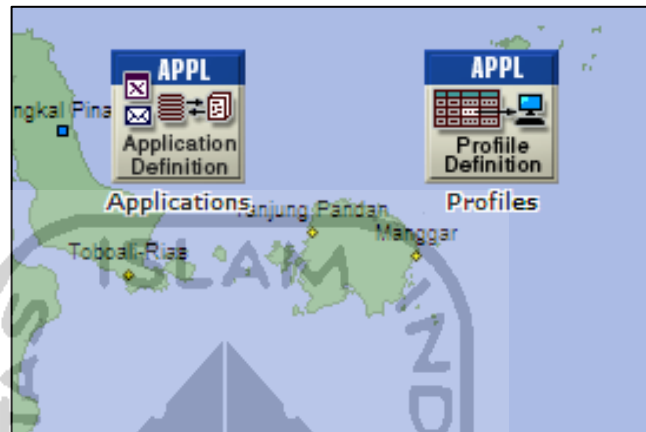
Konfigurasi layanan *web service* pada *server* dapat dilihat pada gambar 3.10.



**Gambar 3.10** Konfigurasi Layanan Web Service



Setelah langkah ini akan muncul **Application Config** dan **Profile Config** secara otomatis. Tampilan dari Application Config dan Profile Config dapat dilihat pada gambar 3.11.



**Gambar 3.11** Application dan Profile Config

Klik kanan pada **Profiles** kemudian **Edit Attributes** → **Profile Configuration**.

**Number of Rows = 0**, kemudian ubah lagi jadi **1**

**Profile Name = Client\_Profile**

Kemudian **Applications** → **Number of Rows = 1** → Pada kolom **Name** pilih **Web Browsing (Heavy Http 1.1)**.

Konfigurasi pada **Profile Config** dapat dilihat pada gambar 3.12.

[-] Profile Configuration	(...)
[-] Number of Rows	1
[-] Client_Profile	
[-] Profile Name	Client_Profile
[-] Applications	(...)
[-] Number of Rows	1
[-] Web Browsing (Heavy HTTP1.1)	
[-] Name	Web Browsing (Heavy HTTP1.1)
[-] Start Time Offset (seconds)	uniform (5,10)
[-] Duration (seconds)	End of Profile
[-] Repeatability	Unlimited

**Gambar 3.12** Konfigurasi Profile Untuk Web Service

## 8. Konfigurasi Permintaan Layanan Pada *Client*

Untuk *client* di Makasar dilakukan konfigurasi permintaan layanan pada *server* dengan cara, klik kanan pada **client** kemudian **Edit Attributes**. Pada **Application: Supported Profiles** ubah nilai **Number of Rows** menjadi **1**.

**Profile Name = Client\_Profile**

**Number of Clients = 1**

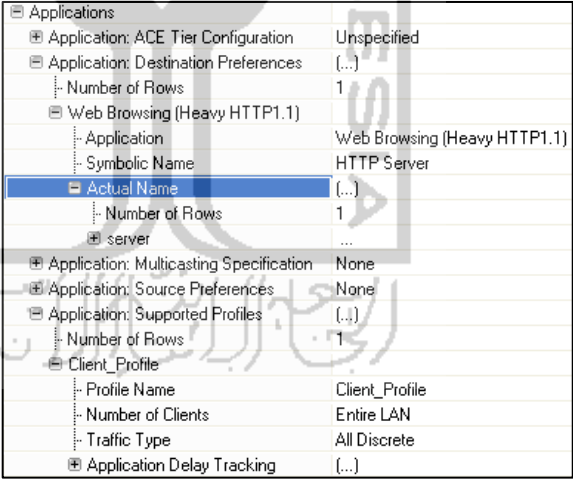
Kemudian pada **Application Destination Preferences** → **Number of Rows = 1**

**Application = Web Browsing (Heavy Http 1.1)**

**Symbolic Name = HTTP Server**

**Actual Name → Name = Server**

Konfigurasi *request* layanan web pada *client* dapat dilihat pada gambar 3.13.



Applications	
Application: ACE Tier Configuration	Unspecified
Application: Destination Preferences	(...)
Number of Rows	1
Web Browsing (Heavy HTTP1.1)	
Application	Web Browsing (Heavy HTTP1.1)
Symbolic Name	HTTP Server
Actual Name	(...)
Number of Rows	1
server	...
Application: Multicasting Specification	None
Application: Source Preferences	None
Application: Supported Profiles	(...)
Number of Rows	1
Client_Profile	
Profile Name	Client_Profile
Number of Clients	Entire LAN
Traffic Type	All Discrete
Application Delay Tracking	(...)

**Gambar 3.13** Konfigurasi Permintaan Web Service Pada Client

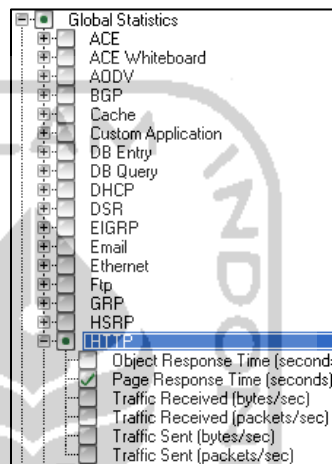
## 9. Penentuan Parameter Yang Akan Diamati

Parameter yang akan diamati pada skenario OSPF\_Http yaitu, waktu respon halaman web, dan trafik yang diterima pada sisi klien. Konfigurasi akan dilakukan secara global maupun pada *node client*.

a. Konfigurasi Global

Klik kanan pada area kerja *project*, pilih **Choose Individual DES Statistics**. Centang pada **Global Statistics** → **HTTP** → **Page Response Time (seconds)**.

Konfigurasi global pada skenario OSPF\_Http dapat dilihat pada gambar 3.14.

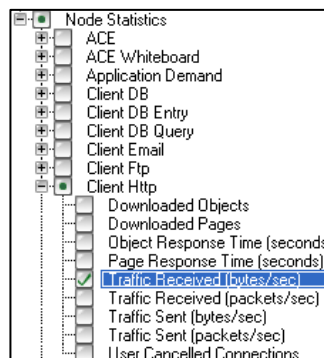


**Gambar 3.14** Parameter Pengamatan Global OSPF\_http

b. Konfigurasi Pada *Client*

Klik kanan pada *client*, pilih **Choose Individual DES Statistics**. Pada **Node Statistics** → **Client Http** → centang pada **Traffic Received (bytes/sec)**.

Konfigurasi *client* pada skenario OSPF\_Http dapat dilihat pada gambar 3.15.

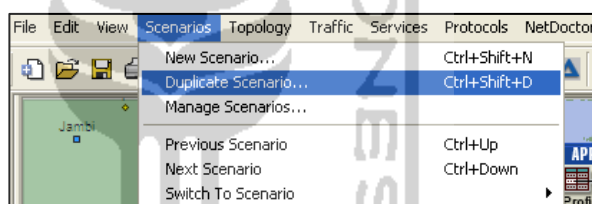


**Gambar 3.15** Parameter Pengamatan Node OSPF\_Http

### 3.4.2 Skenario BGP\_Http

Pada skenario ini diimplementasikan *routing protocol* EBGp pada *router* yang langsung bersinggungan dengan *router* dari AS lain dan IBGP pada *router* internal AS. Pada skenario kedua ini tidak terdapat perbedaan topologi dan parameter pengamatan dengan skenario OSPF\_http, jadi hanya dilakukan duplikasi skenario pertama untuk kemudian diimplementasikan *routing protocol* BGP.

1. Masih pada skenario OSPF\_Http, pilih menu **Scenarios → Duplicate Scenario...** kemudian isikan nama skenario yang baru yaitu BGP\_Http, **Ok**. Proses duplikasi skenario dapat dilihat pada gambar 3.16.



Gambar 3.16 Duplikasi Skenario

2. Konfigurasi BGP

Pada *project* ini akan dikonfigurasi *routing protocol* BGP sebagai EGP dan OSPF dan IBGP sebagai IGP. IBGP dapat berjalan bersamaan dengan OSPF dikarenakan IBGP bukanlah *routing protocol* yang bekerja layaknya *routing protocol* lainnya. IBGP bekerja seperti aplikasi yang mengalir melalui protokol TCP. Klik kanan pada *router*, kemudian **Edit Attributes**.

**IP Routing Protocols → BGP Parameters**

**Status = Enabled**

**Address Family Parameters → Number of Rows = 1**

Buka row 1 tadi → **Address Family = IPv4**

Konfigurasi IPv4 pada *routing protocol* BGP dapat dilihat pada gambar 3.17.

IP Routing Protocols	
BGP Parameters	(...)
Status	Enabled
Start Time	constant (70)
Templates	None
Address Family Parameters	(...)
Number of Rows	1
IPv4	
Address Family	IPv4
Address Family Type	Any
VRF Name	None
Address Family Properties	(...)

**Gambar 3.17** Konfigurasi IPv4 Pada BGP

Langkah selanjutnya yaitu pendistribusian BGP melalui OSPF.

**Address Family Properties → Redistribution → Routing Protocols → OSPF → Redistribute w/ Default.**

Untuk konfigurasi pendistribusian BGP melalui OSPF dapat dilihat pada gambar 3.18.

Address Family Properties	(...)
Synchronization	Enabled
Network Reachability Infor...	None
Network Weight Configurat...	Not Used
Multipath Routes Threshold	1
Redistribution	(...)
Default Metric	Not Configured
Routing Protocols	(...)
Directly Connected	No Redistribution
Static	No Redistribution
EIGRP	No Redistribution
IGRP	No Redistribution
IS-IS	No Redistribution
OSPF	Redistribute w/ Default
RIP	No Redistribution
RIPng	No Redistribution

**Gambar 3.18** Distribusi BGP Melalui OSPF

3. Agar tabel *routing* dapat terlihat jelas, maka perlu dilakukan proses *export* tabel *routing* dari setiap *router*. Langkah – langkahnya masih

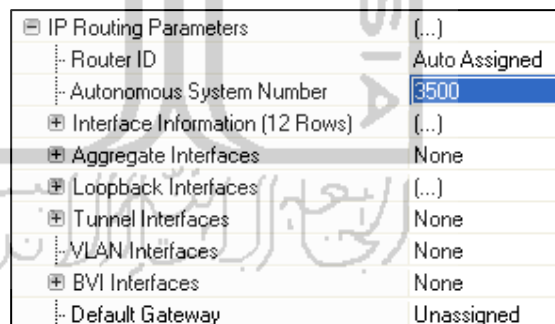
pada **Edit Attributes** → **Reports** → **BGP Routing Table = Export at End of Simulation.**

4. Pada topologi ini akan dibagi menjadi 5 AS. Rincian masing – masing AS dapat dilihat pada tabel 3.3.

**Tabel 3.3** Rincian AS Number

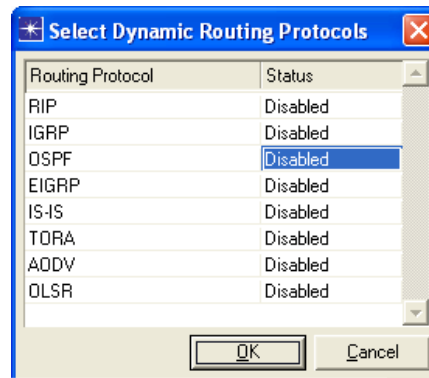
AS Number	Router
2500	router_jkt
3500	router_smg_1, router_smg_2, dan router_smg_3
4500	router_sby_1, dan router_sby_2
5500	router_bjm_1, dan router_bjm_2
6500	router_mkrs

Konfigurasi AS *number* pada *router* dengan cara klik kanan pada *router*, kemudian **Edit Attributes** → **IP** → **IP Routing Parameters**. **Autonomous System Number** diisi sesuai dengan tabel 3.3. Konfigurasi AS *number* dapat dilihat pada gambar 3.19.



**Gambar 3.19** Konfigurasi AS Number

5. Untuk mengaktifkan BGP pada *interface* yang bersinggungan langsung dengan *router* pada AS lain caranya yaitu, **IP** → **IP Routing Parameters** → **Interface Information** → pilih *interface* yang dimaksud (sebagai contoh IF10 pada router\_jkt), pada **Routing Protocols** diaktifkan opsi **Disabled** pada **OSPF**. Konfigurasi untuk menonaktifkan OSPF dapat dilihat pada gambar 3.20.



**Gambar 3.20** Aktivasi Routing Protocol BGP

6. Langkah selanjutnya yaitu mendefinisikan BGP *neighbor* pada setiap *router* agar BGP berjalan. Parameter *neighbor* disini yaitu *ip address*, *AS*, dan juga *update source*. Jika digunakan untuk EBGP, *ip address* dan *remote AS* yang dimaksud adalah *ip address* dari *router AS* lain yang *directly connected*. Sedangkan jika digunakan untuk IBGP, *ip address* yang dimaksud yaitu *ip address router* pada satu *AS (directly connected)* yang sama dengan menggunakan *interface loopback*. Konfigurasi BGP *neighbor* pada OPNET dapat dilihat pada tabel 3.4.

**Tabel 3.4** Parameter BGP Neighbor

Router	Row 0	Row 1	Row 2	Row 3
router_jkt	IP Address : 192.0.1.2 Remote AS : 3500 Update Source : Not Used	IP Address : 192.0.2.2 Remote AS : 3500 Update Source : Not Used		
router_smg _1	IP Address : 192.0.1.1 Remote AS : 2500 Update Source : Not Used	IP Address : 192.0.10.1 Remote AS : 3500 Update Source : Loopback		
router_smg _2	IP Address : 192.0.2.1 Remote AS : 2500 Update Source : Not Used	IP Address : 192.0.10.1 Remote AS : 3500 Update Source : Loopback		

router_smg_3	IP Address : 192.0.8.2 Remote AS : 4500 Update Source : Not Used	IP Address : 192.0.9.2 Remote AS : 4500 Update Source : Not Used	IP Address : 192.0.5.1 Remote AS : 3500 Update Source : Loopback	IP Address : 192.0.7.1 Remote AS : 3500 Update Source : Loopback
router_sby_1	IP Address : 192.0.8.1 Remote AS : 3500 Update Source : Not Used	IP Address : 192.0.14.1 Remote AS : 4500 Update Source : Loopback		
router_sby_2	IP Address : 192.0.9.1 Remote AS : 3500 Update Source : Not Used	IP Address : 192.0.13.2 Remote AS : 5500 Update Source : Not Used	IP Address : 192.0.12.1 Remote AS : 4500 Update Source : Loopback	
router_bjm_1	IP Address : 192.0.13.1 Remote AS : 4500 Update Source : Not Used	IP Address : 192.0.15.2 Remote AS : 6500 Update Source : Not Used	IP Address : 192.0.19.1 Remote AS : 5500 Update Source : Loopback	
router_bjm_2	IP Address : 192.0.18.2 Remote AS : 6500 Update Source : Not Used	IP Address : 192.0.17.1 Remote AS : 5500 Update Source : Loopback		
router_mkstr	IP Address : 192.0.15.1 Remote AS : 5500 Update Source : Not Used	IP Address : 192.0.18.1 Remote AS : 5500 Update Source : Not used		

Langkah pengerjaannya yaitu klik kanan pada *router*, **Edit attributes** → **IP Routing Protocols** → **BGP Parameters** → **Neighbor Information**. Untuk jumlah **Row** disesuaikan dengan jumlah *neighbor* dari tiap *router* (termasuk *interface* loopback untuk iBGP). Kemudian isi semua parameter yang ada pada **IP Address**, **Remote AS**, dan untuk **Update Source** ada pada herarki **Neighbor Properties**. Kemudian pada **Neighbor Properties** → **Address**



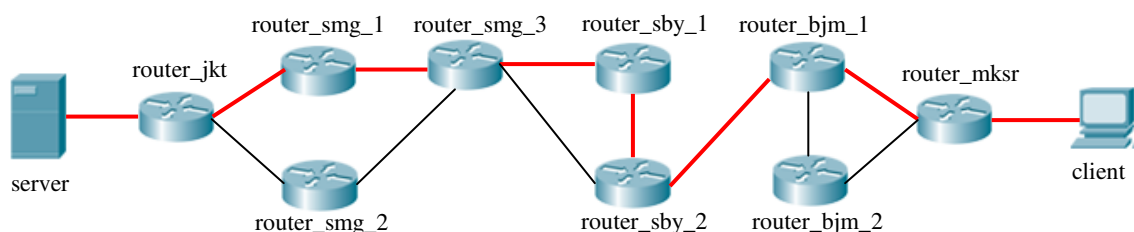
**Family Parameters** → **Number of Rows** isi dengan **1** → **Not Configured** → **Address Family** isi dengan **IPv4**. Konfigurasi BGP *neighbor* dapat dilihat pada gambar 3.21.

Neighbors	[...]
Number of Rows	2
192.0.1.1	...
192.0.10.1	...
IP Address	192.0.10.1
Remote AS	3500
Neighbor Properties	[...]
Address Family Parameters	[...]
Number of Rows	1
IPv4	[...]
Address Family	IPv4
Address Family Type	Any
VRF Name	None
Address Family Properties	[...]
Cluster ID	Same as Router-Wide Setting
Default Metric	[...]
EBGP Multihop	No EBGP Multihop
Next Hop Self	Default
Update Source	LB0

**Gambar 3.21** Konfigurasi Pada BGP Neighbor

### 3.4.3 Skenario BGP\_Policy\_Http

Selanjutnya yaitu membuat skenario ketiga yang penulis beri nama BGP\_Policy\_Http. Pada skenario ini diterapkan sebuah *policy*/kebijakan yang berkaitan dengan dua atribut BGP, yaitu AS Path dan Local Preference. Sebelumnya penulis mengamati *throughput* dari skenario ke dua dan mengamati rute dari *server* menuju *client*. Jalur ditandai dengan garis warna merah pada gambar 3.22.



**Gambar 3.22** Jalur Dari Server Menuju Client Pada BGP\_Simple

Jalur yang akan dimodifikasi yaitu jalur router\_smg\_3 <-> router\_sby\_1. Nilai Local Preference dari jalur ini menuju AS 6500 (router\_mksr) akan diubah menjadi lebih kecil, sehingga nantinya yang akan dipilih sebagai jalur oleh router\_smg\_3 adalah jalur router\_smg\_3 <-> router\_sby\_2. Secara *default*, nilai Local Preference adalah 100. Sebelumnya penulis lakukan duplikasi pada skenario BGP\_Http, dan mengganti nama skenario yang baru menjadi BGP\_Policy\_Http.

1. Masih pada skenario BGP\_Http, pilih menu **Scenarios → Duplicate Scenario...** kemudian penulis isikan nama skenario yang baru yaitu BGP\_Policy\_Http, **Ok**.
2. Idenya yaitu mengurangi derajat preferensi (Local Preference) untuk menuju ke AS 6500. Klik kanan pada router\_smg\_3, **Edit Attributes → IP → IP Routing Parameters → Route Map Configuration → Number of Rows** isi dengan **1** kemudian buka row yang baru dibuat tadi → **Map Label** pilih **Route Map 1**. Pada **Map Configuration →** penulis set **Number of Rows** menjadi **1**, buka row tersebut → pada **Match Info**, **Number of Rows** diset menjadi **1**, buka row tersebut, kemudian penulis mengganti beberapa parameter berikut :
  - a. **Match Property = AS Path**
  - b. **Match Condition = Contains**
  - c. **Match Value = 6500**

Kemudian pada **Set Info →** penulis set nilai **Number of Rows** menjadi **1**. Setelah itu penulis mengganti beberapa parameter berikut :

- a. **Set Attribute = Local Preference**
- b. **Set Operation = Set As [=]**
- c. **Set Value = 25**

Konfigurasi untuk mengurangi derajat preferensi dari jalur router\_smg\_3 <-> router\_sby\_1 dapat dilihat pada gambar 3.23.

Route Map Configuration	(...)
Number of Rows	1
Route Map 1	
Map Label	Route Map 1
Map Configuration	(...)
Number of Rows	1
10	
Term	10
Match Info	(...)
Number of Rows	1
AS Path	
Match Property	AS Path
Match Condition	Contains
Match Value	6500
Set Info	(...)
Number of Rows	1
Local Preference	
Set Attribute	Local Preference
Set Operation	Set As [=]
Set Value	25
Action	Permit

**Gambar 3.23** Modifikasi Atribut Local Preference

3. Langkah selanjutnya yaitu mengimplementasikan *policy* yang baru saja dibuat ke *neighbor* BGP. Pada **IP Routing Protocols** → **BGP Parameters** → **Neighbors** → **192.0.8.2** (ip address remote router yang jalurnya dipilih sebagai rute pada router\_smg\_3) → **Neighbors Properties** → **Address Family Parameters** → **IPv4** → **Address Family Properties** → **Routing policies** → **Number of Rows** penulis set menjadi **1** → buka row yang baru dibuat tadi, pada **Route Map / Route Policy** pilih **Route Map 1**, **Applicable Direction** pilih **In**. Pengimplementasian *policy* pada salah satu *neighbor* BGP dapat dilihat pada gambar 3.24.

192.0.8.2	
IP Address	192.0.8.2
Remote AS	4500
Neighbor Properties	(...)
Address Family Parameters	(...)
Number of Rows	1
IPv4	
Address Family	IPv4
Address Family Type	Any
VRF Name	None
Address Family Properties	(...)
Capability ORF Pre...	None
Default Information	Do Not Originate
Maximum Prefix	(...)
Next Hop Self	Disabled
Remove Private AS	Disabled
Routing Policies	(...)
Number of Rows	1
Route Map 1	
Route Map / ...	Route Map 1
Applicable Dir...	In

**Gambar 3.24** Implementasi Policy Pada Neighbor BGP

### 3.4.4 Skenario OSPF\_Voice, BGP\_Voice, dan BGP\_Policy\_Voice

Konfigurasi pada ketiga skenario ini hampir seluruhnya merupakan duplikasi dari ketiga skenario yang telah dibuat diatas, yaitu OSPF\_Http, BGP\_Http, dan BGP\_Policy\_Http. Yang dibedakan yaitu pada konfigurasi layanan yang didistribusikan dari *server* menuju *client* dan parameter pengamatan.

#### 1. Konfigurasi Layanan Pada *Server*

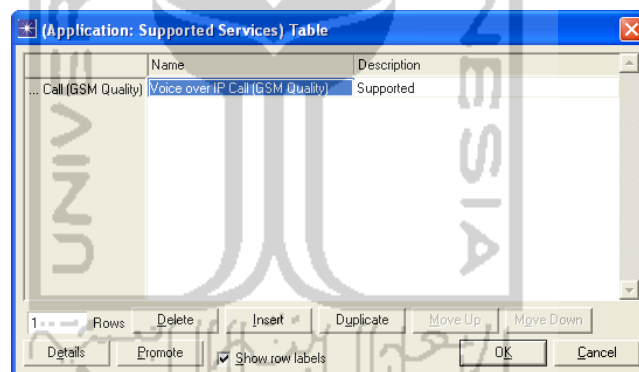
Pada skenario ketiga skenario ini, layanan yang akan didistribusikan oleh *server* yaitu VOIP. Untuk mengaktifkan layanan ini pada *server*, klik kanan pada **server**, kemudian **Edit Attributes**.

**Application: Supported Services → Edit**

**Name = Voice over IP Call (GSM Quality)**

**Description = Supported**

Konfigurasi layanan VOIP pada *server* dapat dilihat pada gambar 3.25.



**Gambar 3.25** Konfigurasi Layanan VOIP

Klik kanan pada **Profiles** kemudian **Edit Attributes → Profile Configuration**.

**Number of Rows = 1**

**Profile Name = Client\_Profile**

Kemudian **Applications → Number of Rows = 1 →** Pada kolom **Name** pilih **Voice over IP Call (GSM Quality)**.

Konfigurasi parameter VOIP dapat dilihat pada gambar 3.26.

[-] Profile Configuration	[...]
Number of Rows	1
[-] Client_Profile	
Profile Name	Client_Profile
[-] Applications	[...]
Number of Rows	1
[-] Voice over IP Call (GSM Quality)	
Name	Voice over IP Call (GSM Quality)
Start Time Offset (seconds)	uniform (5,10)
Duration (seconds)	End of Profile
[-] Repeatability	[...]
Operation Mode	Serial (Ordered)
Start Time (seconds)	uniform (100,110)
Duration (seconds)	End of Simulation
[-] Repeatability	Once at Start Time

**Gambar 3.26** Konfigurasi Profile Untuk VOIP

## 2. Konfigurasi Permintaan Layanan Pada *Client*

Untuk *client* di Makassar, dilakukan konfigurasi permintaan layanan pada *server* dengan cara, klik kanan pada **client** kemudian **Edit Attributes**. Pada **Application: Supported Profiles** ubah nilai **Number of Rows** menjadi **1**.

**Profile Name = Client\_Profile**

**Number of Clients = 1**

Kemudian **Application Destination Preferences** → **Number of Rows = 1**

**Application = Voice over IP Call (GSM Quality)**

**Symbolic Name = Voice Destination**

**Actual Name** → **Name = Server**

Konfigurasi permintaan layanan VOIP pada *client* di Makassar dapat dilihat pada gambar 3.27.

[-] Application: Destination Preferences	(...)
Number of Rows	1
[-] All Applications	
Application	All Applications
Symbolic Name	Voice Destination
[-] Actual Name	(...)
Number of Rows	1
server	...
Application: Multicasting Specification	None
Application: Source Preferences	None
[-] Application: Supported Profiles	(...)
Number of Rows	1
[-] Client_Profile	
Profile Name	Client_Profile
Number of Clients	Entire LAN
Traffic Type	All Discrete
Application Delay Tracking	(...)
Application: Supported Services	None
Application: Transport Protocol	Default

**Gambar 3.27** Konfigurasi Permintaan VOIP Pada Klien

### 3. Penentuan Parameter Yang Akan Diamati

Parameter yang akan diamati pada ketiga skenario ini yaitu, variasi *delay* pada VOIP, dan trafik yang diterima pada sisi *client*. Konfigurasi akan dilakukan secara global maupun pada *node client*.

#### a. Konfigurasi Global

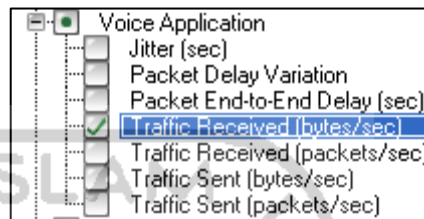
Klik kanan pada area kerja *project*, pilih **Choose Individual DES Statistics**. Centang pada **Global Statistics** → **VOIP** → **Page Response Time (seconds)**. Konfigurasi pengamatan global dapat dilihat pada gambar 3.28.

[-] <input checked="" type="checkbox"/> Voice
<input type="checkbox"/> Jitter (sec)
<input type="checkbox"/> MOS Value
<input checked="" type="checkbox"/> Packet Delay Variation
<input type="checkbox"/> Packet End-to-End Delay (sec)
<input type="checkbox"/> Traffic Received (bytes/sec)
<input type="checkbox"/> Traffic Received (packets/sec)
<input type="checkbox"/> Traffic Sent (bytes/sec)
<input type="checkbox"/> Traffic Sent (packets/sec)

**Gambar 3.28** Parameter Pengamatan Global VOIP

b. Konfigurasi Pada *Client*

Klik kanan pada **client**, pilih **Choose Individual DES Statistics**. Pada **Node Statistics** → **Voice Application** → centang pada **Traffic Received (bytes/sec)**. Konfigurasi pengamatan pada *node client* dapat dilihat pada gambar 3.29.



Gambar 3.29 Parameter Pengamatan Node VOIP

### 3.4.5 Skenario OSPF\_Video, BGP\_Video, dan BGP\_Policy\_Video

Konfigurasi pada ketiga skenario ini hampir seluruhnya merupakan duplikasi dari ketiga skenario yang telah dibuat diatas, yaitu OSPF\_Http, BGP\_Http, dan BGP\_Policy\_Http. Yang dibedakan yaitu pada konfigurasi layanan yang didistribusikan dari *server* menuju *client* dan parameter pengamatan.

1. Konfigurasi Layanan Pada *Server*

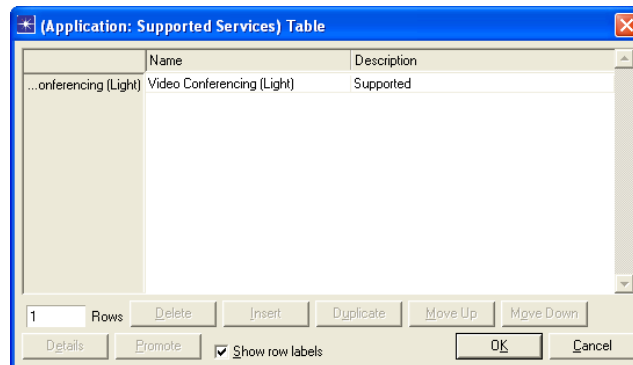
Pada skenario ketiga skenario ini, layanan yang akan didistribusikan oleh *server* yaitu *video conference*. Untuk mengaktifkan layanan ini pada *server*, klik kanan pada **server**, kemudian **Edit Attributes**.

**Application: Supported Services** → **Edit**

**Name = Video Conferencing (Light)**

**Description = Supported**

Konfigurasi layanan video pada *server* dapat dilihat pada gambar 3.30.



**Gambar 3.30** Konfigurasi Layanan Video Conference

Klik kanan pada **Profiles** kemudian **Edit Attributes** → **Profile Configuration**.

**Number of Rows = 1**

**Profile Name = Client\_Profile**

Kemudian **Applications** → **Number of Rows = 1** → Pada kolom **Name** pilih **Video Conferencing (Light)**.

Konfigurasi parameter layanan video dapat dilihat pada gambar 3.31.

[-] Profile Configuration	(...)
Number of Rows	1
[-] Client_Profile	
Profile Name	Client_Profile
[-] Applications	(...)
Number of Rows	1
[-] Video Conferencing (Light)	
Name	Video Conferencing (Light)
Start Time Offset (seconds)	uniform (5,10)
Duration (seconds)	End of Profile
[-] Repeatability	Unlimited
Operation Mode	Serial (Ordered)
Start Time (seconds)	uniform (100,110)
Duration (seconds)	End of Simulation
[-] Repeatability	Once at Start Time

**Gambar 3.31** Konfigurasi Profile Untuk Video Conference

## 2. Konfigurasi Permintaan Layanan Pada *Client*

Untuk *client* di Makassar dilakukan konfigurasi permintaan layanan pada *server* dengan cara, klik kanan pada **client** kemudian **Edit Attributes**.



Pada **Application: Supported Profiles** ubah nilai **Number of Rows** menjadi **1**.

**Profile Name = Client\_Profile**

**Number of Clients = 1**

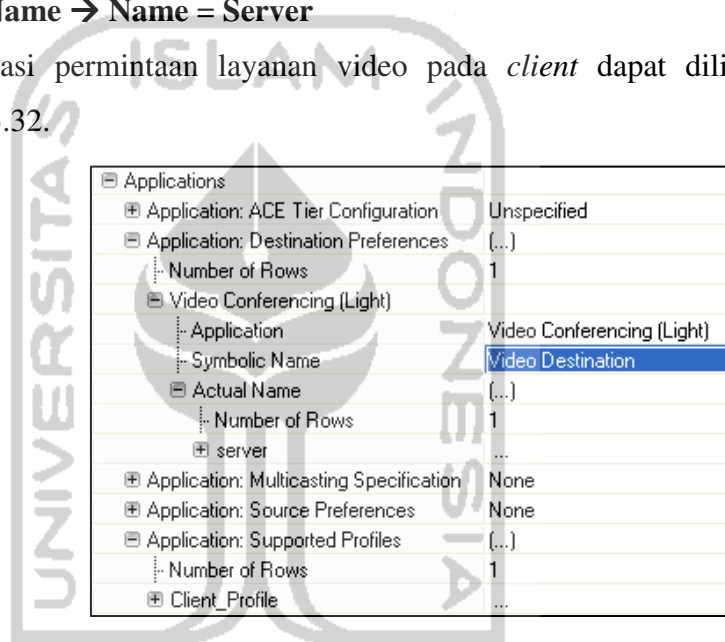
Kemudian **Application Destination Preferences** → **Number of Rows = 1**

**Application = Video Conferencing (Light)**

**Symbolic Name = Video Destination**

**Actual Name** → **Name = Server**

Konfigurasi permintaan layanan video pada *client* dapat dilihat pada gambar 3.32.



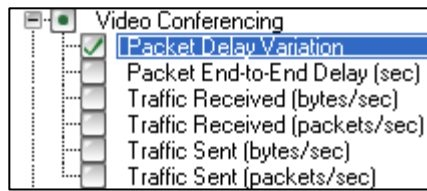
**Gambar 3.32** Konfigurasi Permintaan Video Conference Pada Klien

### 3. Penentuan Parameter Yang Akan Diamati

Parameter yang akan diamati pada ketiga skenario ini yaitu, variasi *delay* pada VOIP, dan trafik yang diterima pada sisi klien. Konfigurasi akan dilakukan secara global maupun pada *node client*.

#### a. Konfigurasi Global

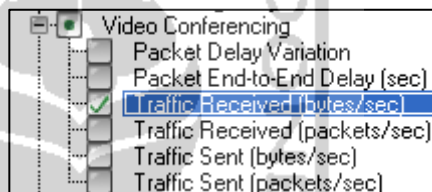
Klik kanan pada area kerja *project*, pilih **Choose Individual DES Statistics**. Centang pada **Global Statistics** → **VOIP** → **Page Response Time (seconds)**. Konfigurasi pengamatan global dapat dilihat pada gambar 3.33.



**Gambar 3.33** Parameter Pengamatan Global Video Conference

b. Konfigurasi Pada *Client*

Klik kanan pada **client**, pilih **Choose Individual DES Statistics**. Pada **Node Statistics** → **Video Conferencing** → centang pada **Traffic Received (bytes/sec)**. Konfigurasi pengamatan pada *node client* dapat dilihat pada gambar 3.34.



**Gambar 3.34** Parameter Pengamatan Node Video Conference

### 3.4.6 Skenario OSPF\_Simple, BGP\_Simple, dan BGP\_Policy

Konfigurasi pada ketiga skenario ini hampir seluruhnya merupakan duplikasi dari ketiga skenario yang telah dibuat diatas, yaitu OSPF\_Http, BGP\_Http, dan BGP\_Policy\_Http, yang membedakan yaitu tidak adanya layanan yang didistribusikan dari *server* menuju *client*. Sehingga trafik yang dihasilkan hanya merupakan trafik dari *router*.

1. Konfigurasi Layanan Pada *Server*

Pada ketiga skenario ini, tidak ada layanan yang akan didistribusikan oleh *server*. Untuk menonaktifkan layanan ini pada *server*, klik kanan pada **server**, kemudian **Edit Attributes**.

**Application: Supported Services** → **None**

Konfigurasi untuk menonaktifkan layanan pada *server* dapat dilihat pada gambar 3.35.

name	server
Applications	
Application: ACE Tier Configuration	Unspecified
Application: Destination Preferences	None
Application: Supported Profiles	None
Application: Supported Services	None
CPU	
VPN	

**Gambar 3.35** Konfigurasi Penonaktifan Layanan Pada Server

Karena pada skenario ini tidak membutuhkan layanan tertentu, maka objek **Profile** dan **Application** dihapus saja.

## 2. Konfigurasi Permintaan Layanan Pada *Client*

Untuk *client* di Makassar, dilakukan konfigurasi untuk menonaktifkan permintaan layanan pada *server* dengan cara, klik kanan pada **client** kemudian **Edit Attributes**. Pada **Application: Supported Profiles** ubah nilai **Number of Rows** menjadi **0**. Kemudian pada **Application Destination Preferences** → **Number of Rows** = **0**. Konfigurasi untuk menonaktifkan permintaan layanan pada *client* dapat dilihat pada gambar 3.36.

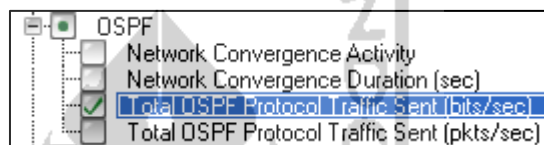
name	client
Applications	
Application: ACE Tier Configuration	Unspecified
Application: Destination Preferences	(...)
Number of Rows	0
Application: Multicasting Specification	None
Application: Source Preferences	None
Application: Supported Profiles	(...)
Number of Rows	0
Application: Supported Services	None
Application: Transport Protocol	Default

**Gambar 3.36** Konfigurasi Penonaktifan Layanan Pada Klien

### 3. Penentuan Parameter Yang Akan Diamati

Parameter yang akan diamati pada ketiga skenario ini yaitu, trafik *routing protocol*, dan penggunaan *resource* CPU. Konfigurasi akan dilakukan secara global. Klik kanan pada area kerja *project*, pilih **Choose Individual DES Statistics**.

- a. Centang pada **Global Statistics** → **OSPF** → **Total OSPF Protocol Traffic Sent (bits/sec)**, untuk skenario OSPF\_Simple. Konfigurasi pengamatan global untuk OSPF\_Simple dapat dilihat pada gambar 3.37.



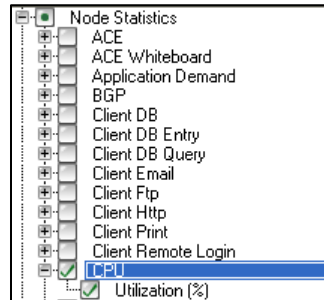
**Gambar 3.37** Pengamatan Trafik OSPF

- b. Centang pada **Global Statistics** → **BGP** → **Traffic Sent (bits/sec)**, untuk skenario BGP\_Simple dan BGP Policy. Konfigurasi pengamatan global untuk BGP\_Simple dan BGP\_Policy dapat dilihat pada gambar 3.38.



**Gambar 3.38** Pengamatan Trafik BGP

- c. Centang pada **Node Statistics** → **CPU** → **Utilizations (%)**, untuk semua skenario. Konfigurasi pengamatan *node* pada semua skenario dapat dilihat pada gambar 3.39.



**Gambar 3.39** Pengamatan Penggunaan Resource CPU

### 3.5 Pengujian Sistem

Pada tahap ini akan dijalankan simulasi pada semua skenario yang ada. Uji coba ini dilakukan dengan menjalankan simulasi pada 12 skenario dan membandingkan hasil dari parameter yang telah ditentukan. Parameter simulasi dapat dilihat pada tabel 3.5.

**Tabel 3.5** Parameter Simulasi

Parameter Simulasi	
Duration	10 minutes
Seed	128
Values per statistic	100
Update interval	500.000 events
Simulation Kernel	Optimized
Profil Start Time	<ul style="list-style-type: none"> <li>- HTTP Default, Uniform (100,110)</li> <li>- Voice Default, Uniform (100,110)</li> <li>- Video Default, Uniform (100,110)</li> </ul>
Profil Duration	End of Simulation

## BAB IV

### HASIL DAN PEMBAHASAN

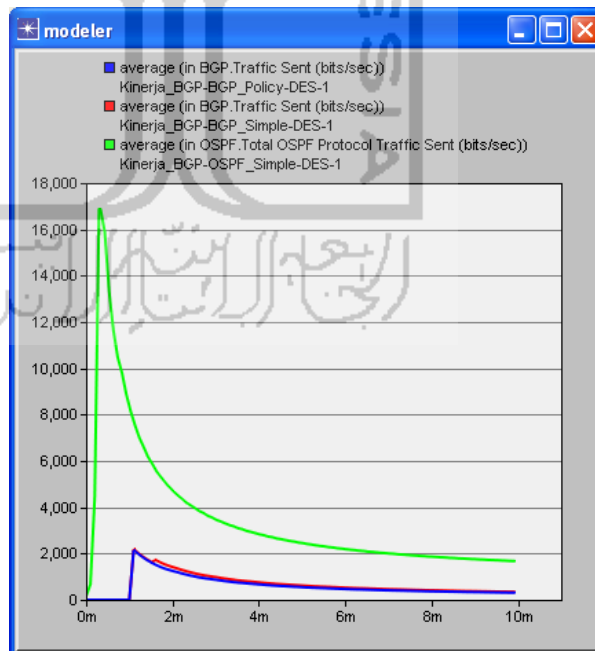
#### 4.1 Analisis dan Evaluasi Hasil Uji Coba

Pada bab ini akan dibahas mengenai hasil analisis dari uji coba *project* Kinerja\_BGP dengan menggunakan simulator OPNET Modeler.

##### 4.1.1 Perbandingan Skenario OSPF\_Simple, BGP\_Simple, dan BGP\_Policy

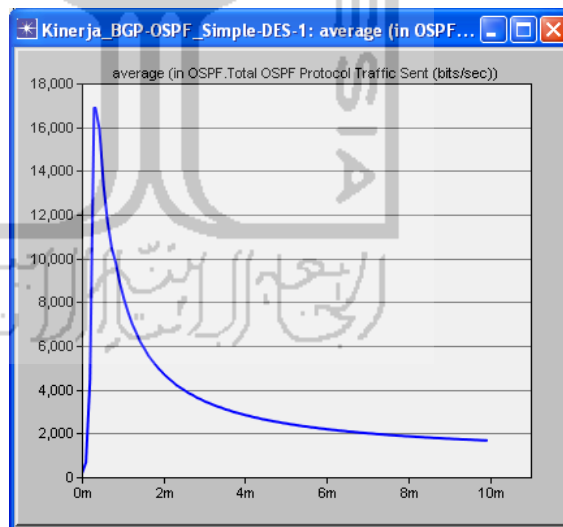
Setelah menjalankan simulasi pada ketiga skenario ini maka didapatkan statistik data berupa grafis yang menunjukkan parameter yang diamati pada setiap skenario. Berikut merupakan beberapa parameter pengamatan yang diamati :

##### 1. Trafik *Routing Protocol*



**Gambar 4.1** Perbandingan Trafik Protokol Routing Terkirim

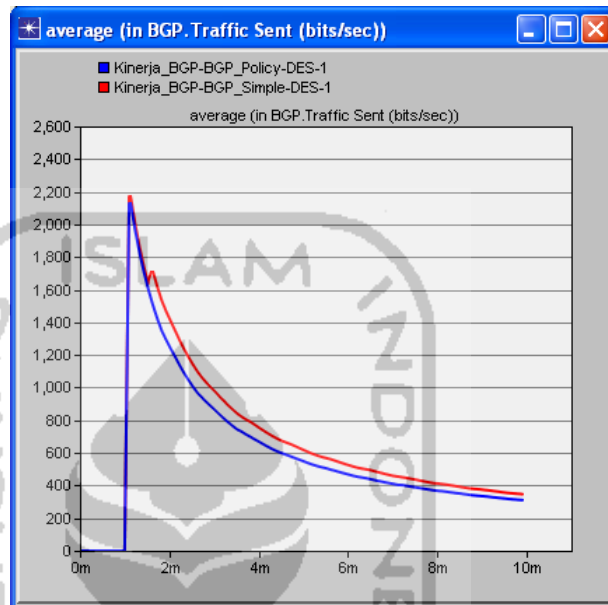
Trafik yang dikirimkan oleh OSPF (hijau) lebih besar dibandingkan dengan trafik BGP (biru dan merah). Pada masa konvergensi, OSPF mengirimkan trafik secara *broadcast* ke semua *node*. OSPF menganut prinsip protokol *link-state* yang terus melakukan *broadcast* ke semua *node* hingga terbentuk topologi utuh dari jaringan dengan area yang sama. Setelah membentuk relasi dengan *router* tetangga (*directly connected*), *router* OSPF akan membuat paket *link-state* yang berisi informasi mengenai tetangga tadi. Kemudian setiap *router* OSPF akan melakukan *flooding* paket *link-state* ke semua tetangganya hingga terbentuk peta jaringan yang utuh. Hal ini yang menyebabkan trafik OSPF jauh lebih tinggi dibandingkan dengan BGP. Maksimal trafik yang didapatkan dari skenario OSPF\_Simple adalah 16918,67 bits/second. Grafik dari total trafik terkirim pada OSPF dapat dilihat pada gambar 4.2.



**Gambar 4.2** Trafik OSPF Terkirim

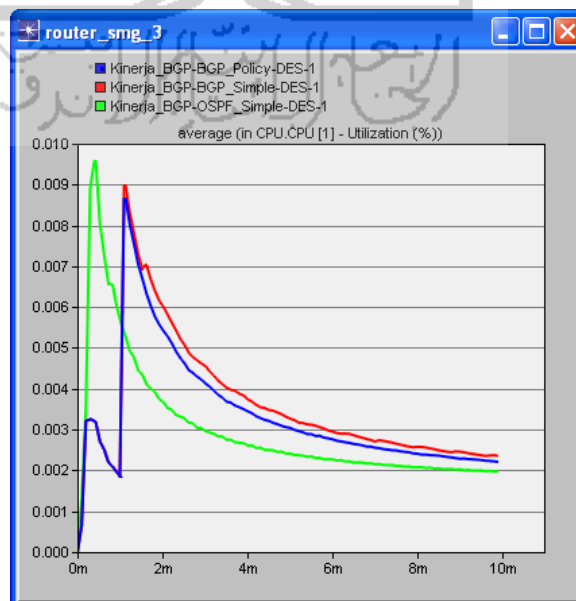
Pada skenario BGP\_Simple dan BGP\_Policy tidak terlihat perbedaan mencolok. Penerapan *policy* tidak menimbulkan perubahan trafik yang signifikan pada BGP. Trafik pada BGP\_Simple terlihat sedikit lebih besar dibandingkan dengan BGP\_Policy. Maksimal trafik pada BGP\_Simple adalah 2179,667 bits/second, sedangkan maksimal

trafik pada BGP\_Policy adalah 2139,444 bits/second. Grafik dari total trafik terkirim pada BGP dan BGP menggunakan *policy* dapat dilihat pada gambar 4.3.



**Gambar 4.3** Perbandingan Trafik BGP\_Simple dan BGP\_Policy

## 2. Penggunaan *Resource* CPU



**Gambar 4.4** Perbandingan Persentase CPU Utilization



Penggunaan *resource* CPU OSPF terlihat lebih banyak walaupun tidak terlalu signifikan dibandingkan dengan BGP dan BGP dengan *policy* dalam skala jaringan yang sama pada router\_smg\_3. Maksimal *resource* CPU yang terpakai pada setiap router pada skenario OSPF\_Simple dapat dilihat pada tabel 4.1.

**Tabel 4.1** Penggunaan CPU OSPF\_Simple

Perangkat	Max CPU Resource (%)
router_jkt	0,005533
router_smg_1	0,004733
router_smg_2	0,0056
router_smg_3	0,0096
router_sby_1	0,004467
router_sby_2	0,0062
router_bjm_1	0,005933
router_bjm_2	0,004533
router_mksr	0,0052
<b>Rata - rata</b>	<b>0,005755</b>

*Resource* CPU yang digunakan pada skenario BGP\_Simple dapat dilihat pada tabel 4.2.

**Tabel 4.2** Penggunaan CPU BGP\_Simple

Perangkat	Max CPU Resource (%)
router_jkt	0,003861
router_smg_1	0,004
router_smg_2	0,004639
router_smg_3	0,009
router_sby_1	0,004444
router_sby_2	0,006278
router_bjm_1	0,005333
router_bjm_2	0,0045
router_mksr	0,003917
<b>Rata - rata</b>	<b>0,005108</b>

*Resource* CPU yang digunakan pada skenario BGP\_Policy dapat dilihat pada tabel 4.3.

**Tabel 4.3** Penggunaan CPU BGP\_policy

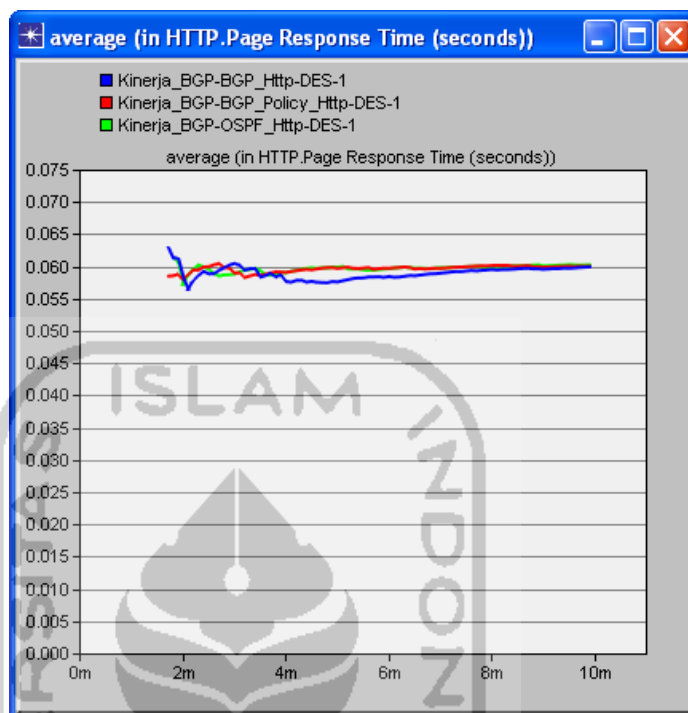
Perangkat	Max CPU Resource (%)
router_jkt	0,003889
router_smg_1	0,004
router_smg_2	0,004667
router_smg_3	0,008694
router_sby_1	0,004528
router_sby_2	0,005889
router_bjm_1	0,005333
router_bjm_2	0,0045
router_mksr	0,003917
<b>Rata - rata</b>	<b>0,005046</b>

Dari ketiga tabel diatas (tabel 4.1, 4.2, dan 4.3) terlihat *routing protocol* yang paling banyak menggunakan *resource* CPU adalah OSPF dibandingkan BGP. Hasil ini menunjukkan bahwa OSPF yang merupakan *link-state protocol* memang membutuhkan *resource* CPU yang besar untuk proses konvergensinya dibandingkan dengan BGP. Sedangkan BGP lebih banyak menggunakan *resource* CPU dibandingkan pada BGP dengan *policy* walaupun tidak terlalu signifikan.

#### 4.1.2 Perbandingan Skenario OSPF\_Http, BGP\_Http, dan BGP\_Policy\_Http

Pada ketiga skenario ini parameter utama yang diamati adalah waktu respon halaman web dan juga trafik yang diterima oleh *client*. Seperti dijelaskan pada bab 3, layanan yang didistribusikan oleh *server* kepada klien nya yaitu *web service*. Berikut merupakan hasil simulasi dari ketiga skenario tersebut :

## 1. Waktu Respon Halaman Web

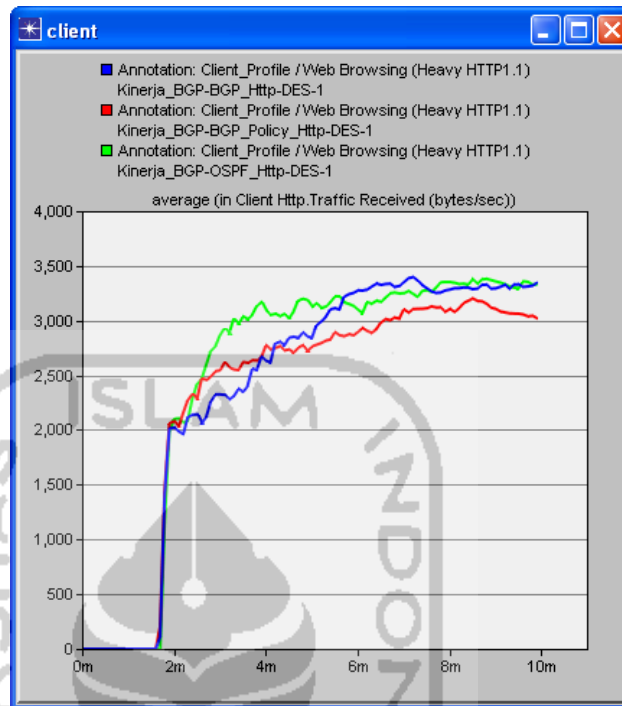


**Gambar 4.5** Perbandingan Rata – Rata Waktu Respon Halaman Web

Perbedaan waktu respon halaman web terlihat hanya pada awal aplikasi web mulai berjalan, kemudian tidak terdapat perbedaan hingga akhir simulasi yaitu pada menit ke-10. BGP\_Policy\_Http (merah) sedikit lebih cepat dibandingkan yang lain dalam merespon halaman web. Waktu respon awal dari BGP\_Http adalah 0,061311607 detik, waktu respon awal dari BGP\_Policy\_Http adalah 0,058541504 detik, sedangkan waktu respon awal dari OSPF\_Http adalah 0,061459 detik. Setelah itu mengalami penurunan kemudian naik lagi. Semakin lama aplikasi berjalan cenderung stabil baik itu pada OSPF, BGP, maupun BGP dengan *policy* dalam merespon halaman web.

Pada OSPF waktu respon halaman web rata – rata yaitu 0,059682 detik. Pada BGP waktu respon halaman web rata – rata 0,058962 detik. Sedangkan pada BGP dengan penggunaan *policy*, waktu respon halaman web rata – rata adalah 0,059621 detik.

## 2. Trafik Web Yang Diterima Klien



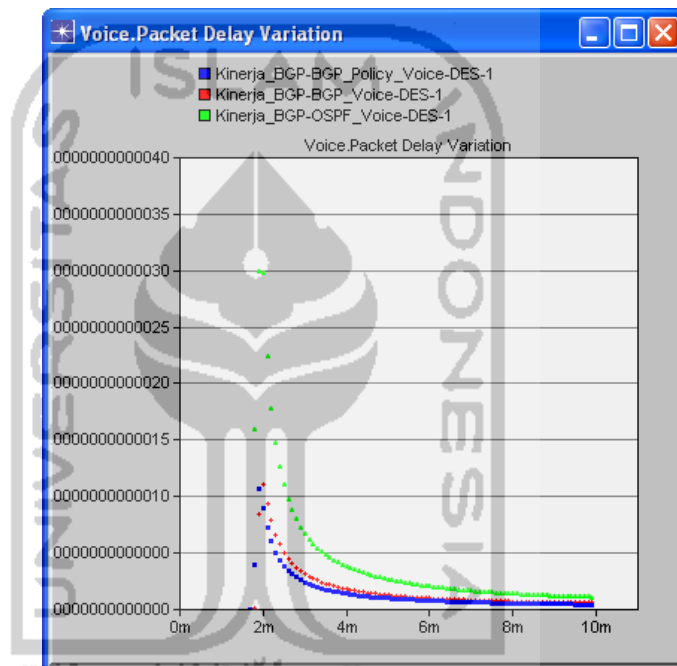
**Gambar 4.6** Perbandingan Trafik Web yang Diterima Klien

Trafik web yang diterima oleh klien pada ketiga skenario ini cenderung mengalami peningkatan. Pada awal jalannya aplikasi web, trafik yang diterima pada skenario OSPF\_Http mengalami peningkatan yang besar dibandingkan skenario lainnya. Pada detik ke-198, klien pada skenario OSPF\_Http sudah mengalami peningkatan trafik hingga 3000 bytes/sec, sedangkan klien pada skenario BGP\_Http 2300 bytes/sec, dan klien pada skenario BGP\_Policy\_Http 2500 bytes/sec. Kemudian hingga simulasi berakhir pada menit ke-10, trafik yang diterima klien pada skenario BGP\_Policy\_http lebih sedikit dibandingkan skenario lainnya, yaitu 3021,795 bytes/sec, skenario BGP\_Http 3344,942 bytes/sec, dan skenario BGP\_OSPF\_Http 3331,103 bytes/sec.

#### 4.1.3 Perbandingan Skenario OSPF\_Voice, BGP\_Voice, dan BGP\_Policy\_Voice

Pada ketiga skenario ini parameter utama yang diamati adalah variasi *delay* dari aplikasi VOIP dan juga trafik VOIP yang diterima oleh klien. Berikut merupakan hasil simulasi dari ketiga skenario tersebut :

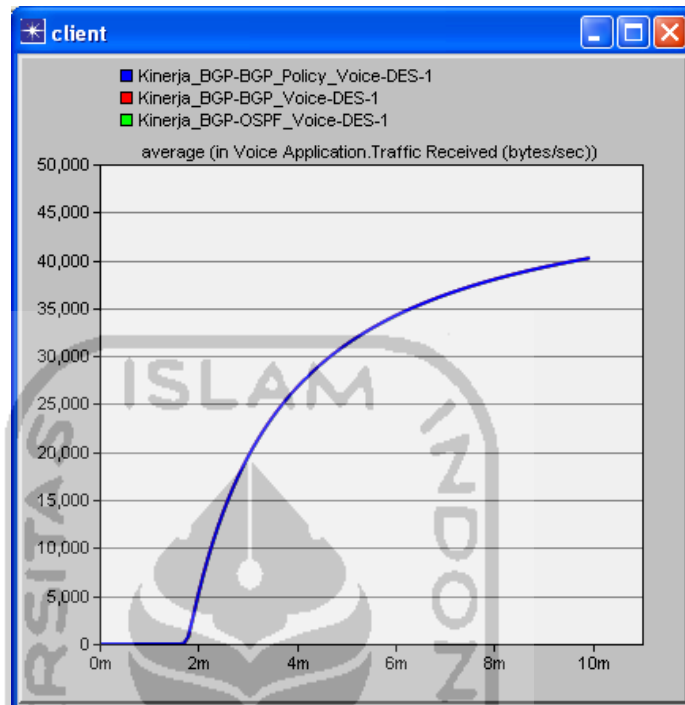
##### 1. Variasi *Delay*



**Gambar 4.7** Perbandingan Variasi Delay Pada VOIP

*Router* dengan *routing protocol* OSPF memiliki variasi *delay* yang paling tinggi dan langsung meningkat pada awal permintaan layanan VOIP oleh klien. Maksimal variasi *delay* dari OSPF adalah  $3 \times 10^{-12}$ , BGP  $1 \times 10^{-12}$ , dan BGP dengan *policy* yaitu  $1 \times 10^{-12}$ . Namun peningkatan ini hanya sesaat, kemudian untuk selanjutnya mengalami penurunan secara terus menerus hingga hampir mendekati nol. Begitu juga dengan *router* BGP dan *router* BGP dengan *policy*. Perbedaan terlihat pada *router* OSPF yang memiliki variasi *delay* lebih tinggi dibandingkan dengan BGP dan BGP dengan *policy* walaupun selisihnya kecil sekali.

## 2. Trafik Voice Yang Diterima Klien



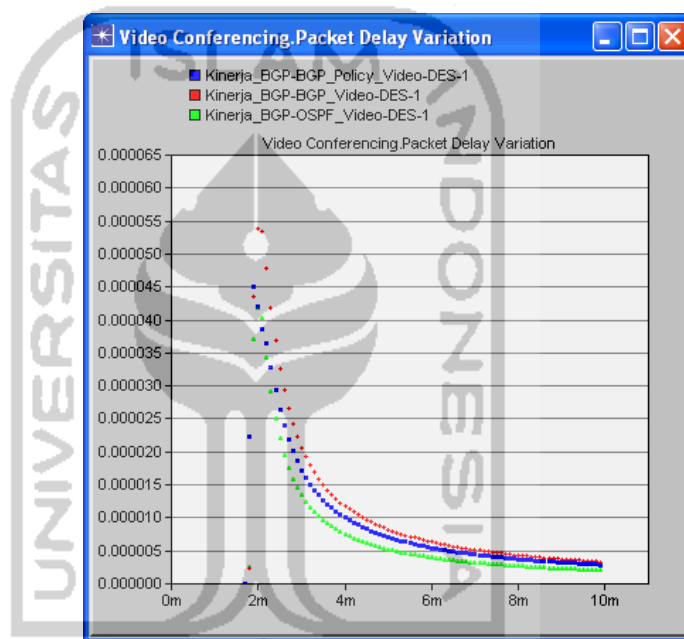
**Gambar 4.8** Perbandingan Trafik VOIP yang Diterima Klien

Tidak terlihat perbedaan pada trafik yang diterima baik itu pada *router* OSPF, BGP, maupun BGP dengan *policy*. Variasi *delay* yang sangat kecil, kurang dari 1ms, sudah bisa dikatakan sangat baik untuk menunjang keperluan VOIP. Sangat kecilnya variasi *delay* ini juga mempengaruhi penerimaan trafik pada klien, dimana bisa dikatakan hampir tidak terdapat perbedaan trafik yang diterima dari sisi klien. Trafik yang diterima cenderung meningkat pada kisaran angka yang sama, dengan trafik akhir yang diterima klien pada menit ke-10 untuk OSPF\_Voice yaitu 40188,225 bytes/sec, untuk BGP\_Voice 40203,295 bytes/sec, dan untuk BGP\_Policy\_Voice yaitu 40202,14 bytes/sec.

#### 4.1.4 Perbandingan Skenario OSPF\_Video, BGP\_Video, dan BGP\_Policy\_Video

Pada ketiga skenario ini parameter utama yang diamati adalah variasi *delay* dari aplikasi *video conference* dan juga trafik video yang diterima oleh klien. Berikut merupakan hasil simulasi dari ketiga skenario tersebut :

##### 1. Variasi *Delay*

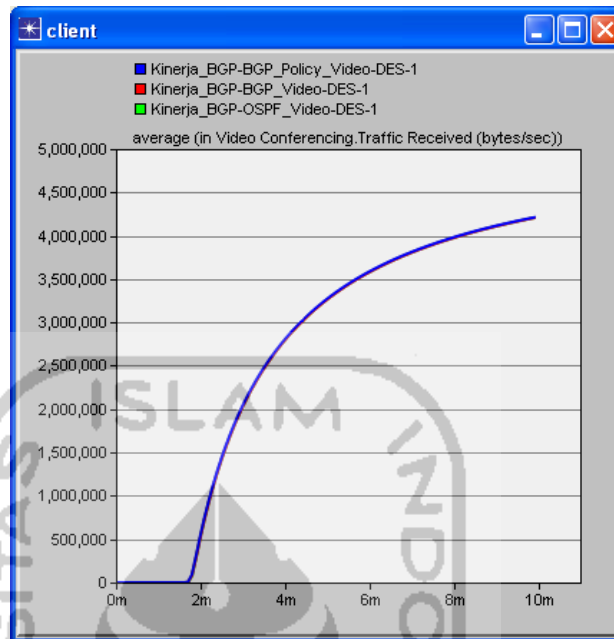


**Gambar 4.9** Perbandingan Variasi Delay Pada Video Conference

*Router* dengan *routing protocol* BGP memiliki variasi *delay* yang paling tinggi pada permintaan layanan *video conference* oleh klien. Sedangkan variasi *delay* paling rendah terjadi pada *router* OSPF. Peningkatan variasi *delay* hanya sesaat terjadi pada awal aplikasi *video conference* berjalan, kemudian untuk selanjutnya stabil mengalami penurunan hingga hampir mendekati nol.

Maksimal variasi *delay* yang terjadi pada *router* BGP adalah  $5,37673 \times 10^{-5}$ , pada *router* BGP dengan *policy* yaitu  $4,51418 \times 10^{-5}$ , dan maksimal variasi *delay* pada *router* OSPF adalah  $4,19185 \times 10^{-5}$ .

## 2. Trafik Video Yang Diterima Klien



**Gambar 4.10** Perbandingan Trafik Video yang Diterima Klien

Sama halnya seperti VOIP, Tidak terlihat perbedaan pada trafik yang diterima baik itu pada *router* OSPF, BGP, maupun BGP dengan *policy*. Variasi *delay* yang sangat kecil, kurang dari 1ms, sudah bisa dikatakan sangat baik untuk menunjang keperluan *video conference*. Sangat kecilnya variasi *delay* ini juga mempengaruhi penerimaan trafik video pada klien, dimana bisa dikatakan hampir tidak terdapat perbedaan trafik yang diterima dari sisi klien. Trafik yang diterima cenderung meningkat pada kisaran angka yang sama dengan trafik akhir yang diterima klien pada menit ke-10 untuk OSPF\_Video yaitu 4208083,2 bytes/sec, untuk BGP\_Video 4207219,2 bytes/sec, dan untuk BGP\_Policy\_Video yaitu 4212201,6 bytes/sec.



## BAB V

### PENUTUP

#### 5.1 Kesimpulan

Berdasarkan hasil penelitian dan pembahasan pada beberapa skenario dari penelitian analisis kinerja *routing protocol* BGP menggunakan BGP *policy*, maka dapat diambil beberapa kesimpulan yaitu sebagai berikut :

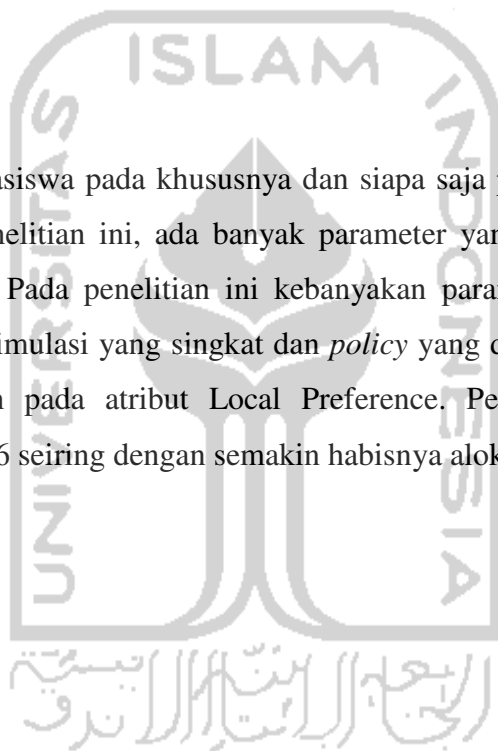
- a. OSPF menghasilkan lebih banyak trafik *routing protocol* dibandingkan dengan BGP dan BGP menggunakan *policy*.
- b. Pada penggunaan *resource* CPU, OSPF lebih banyak memakai *resource* CPU dibandingkan dengan BGP. Dan BGP sedikit lebih banyak mengkonsumsi *resource* CPU dibandingkan dengan BGP menggunakan *policy* walaupun tidak terlalu signifikan.
- c. BGP tanpa *policy* sedikit lebih lambat dalam merespon halaman web dibandingkan BGP dengan *policy* dan OSPF pada awal berjalannya aplikasi, namun mengalami peningkatan dan cenderung lebih cepat dibandingkan dengan BGP dengan *policy* dan OSPF hingga pada akhirnya cenderung sama.
- d. Trafik web yang diterima oleh klien baik itu dari *router* OSPF, BGP, maupun BGP *policy* cenderung stabil dan mengalami peningkatan. Trafik paling sedikit diterima oleh *router* BGP dengan *policy*.
- e. Variasi *delay* VOIP yang terjadi pada *router* OSPF lebih besar dibandingkan BGP dan BGP dengan *policy*. Namun nilai variasi *delay* ini tidak mempengaruhi penerimaan trafik pada klien.
- f. Tidak terlihat perbedaan pada klien di jaringan dengan *router* OSPF, BGP, dan BGP dengan *policy* dalam penerimaan trafik *video*

*conference*. Sama seperti pada layanan VOIP, variasi *delay* tidak mempengaruhi penerimaan trafik pada klien.

- g. Dengan penerapan satu buah *policy* pada satu *router* BGP tidak terlalu mempengaruhi kinerja perangkat antara *router* BGP yang tidak menerapkan *policy* dengan yang menerapkan *policy*. Sebaliknya perbandingan antara OSPF dengan BGP (dan BGP menggunakan *policy*) mempengaruhi kinerja jaringan secara keseluruhan.

## 5.2 Saran

Bagi mahasiswa pada khususnya dan siapa saja pada umumnya yang ingin melanjutkan penelitian ini, ada banyak parameter yang masih bisa dieksplorasi dalam OPNET. Pada penelitian ini kebanyakan parameter masih diatur secara *default*, waktu simulasi yang singkat dan *policy* yang diterapkan hanya satu buah yaitu perubahan pada atribut Local Preference. Perlu dipertimbangkan juga penggunaan IPv6 seiring dengan semakin habisnya alokasi IPv4.



## DAFTAR PUSTAKA

- Andriansyah, Fajar, 2009. Instalasi OPNET Modeler 14.0. (On-line) Available at <http://iesmartlife.wordpress.com/2009/12/17/instalasi-opnet-modeler-14-0/>, di akses tanggal 7 Januari 2012.
- Cisco. BGP Best Path Selection Algorithm. (On-line) Available at [http://www.cisco.com/en/US/tech/tk365/technologies\\_tech\\_note09186a0080094431.shtml](http://www.cisco.com/en/US/tech/tk365/technologies_tech_note09186a0080094431.shtml), diakses tanggal 31 Maret 2012.
- Cisco Networking Academy. 2010. CCNA Exploration 4.0: Network Fundamental. Cisco System, Inc.
- Cisco Networking Academy. 2010. CCNA Exploration 4.0: Routing Protocol and Concepts. Cisco System, Inc.
- IETF, 1994. A Border Gateway Protocol 4 (BGP-4). (On-line) Available at <http://datatracker.ietf.org/doc/rfc1654/>, diakses tanggal 14 Januari 2012.
- Laboratorium Sistem dan Jaringan Komputer. 2010. Modul Praktikum Jaringan Komputer. Yogyakarta: Universitas Islam Indonesia.
- Opnet, 2011. System Requirements. (On-line) Available at [http://www.opnet.com/solutions/system\\_requirements/modeler/](http://www.opnet.com/solutions/system_requirements/modeler/), di akses tanggal 24 Maret 2012.
- Peterson, Larry, & Davie, Bruce. 2008. Computer Networks A System Approach. United State of America: Morgan Kauffman.
- Rafiudin, Rahmat.2004. Multihoming Menggunakan BGP (Border Gateway Protocol). Yogyakarta: Penerbit Andi.
- Sofana, Iwan. 2008. Membangun Jaringan Komputer. Bandung: Informatika.
- Stallings, William, 2003. Data and Computer Communications. New Jersey: Prentice Hall.

Svensson, Tommy, & Popescu, Alex. 2003. Development of laboratory exercises based on OPNET Modeler. (On-line) Available at [http://www.opnet.com/university\\_program/teaching\\_with\\_opnet/text-books\\_and\\_materials/materials/Lab\\_Exercices\\_Modeler.pdf](http://www.opnet.com/university_program/teaching_with_opnet/text-books_and_materials/materials/Lab_Exercices_Modeler.pdf), di akses tanggal 25 Maret 2012.

White, Russ, & McPherson, Danny, & Srihari, Sangli. 2004. Practical BGP. Boston: Pearson Education.

