UNIVERSITAS INDONESIA

SIMULASI DAN ANALISA PENGARUH *TCP WINDOWING* PADA *TRANSPORT LAYER* TERHADAP PENINGKATAN KINERJA JARINGAN BERBASIS *VIRTUAL PRIVATE NETWORK* (VPN) MENGGUNAKAN SIMULATOR OPNET

SKRIPSI

Oleh :

YOMMA HENDRA PUTRA
0606078550

PROGRAM STUDI TEKNIK KOMPUTER
DEPARTEMEN TEKNIK ELEKTRO
FAKULTAS TEKNIK UNIVERSITAS INDONESIA
DEPOK
DESEMBER 2010

UNIVERSITAS INDONESIA

SIMULASI DAN ANALISA PENGARUH *TCP WINDOWING* PADA *TRANSPORT LAYER* TERHADAP PENINGKATAN KINERJA JARINGAN BERBASIS *VIRTUAL PRIVATE NETWORK* (VPN) MENGGUNAKAN SIMULATOR OPNET

SKRIPSI

Diajukan sebagai salah satu syarat untuk memperoleh gelar Sarjana Teknik.

YOMMA HENDRA PUTRA
0606078550

PROGRAM STUDI TEKNIK KOMPUTER
DEPARTEMEN TEKNIK ELEKTRO
FAKULTAS TEKNIK UNIVERSITAS INDONESIA
DEPOK
DESEMBER 2010

HALAMAN PENGESAHAN

Skripsi ini diajukan oleh :

Nama : Yomma Hendra Putra

NPM : 0606078550

Program Studi : Teknik Komputer

Judul Skripsi : Simulasi Dan Analisa Pengaruh *TCP Windowing* Pada *Transport Layer* Terhadap Peningkatan Kinerja Jaringan Berbasis *Virtual Private Network* (VPN) Menggunakan Simulator OPNET

Telah berhasil dipertahankan di hadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Sarjana Teknik pada Program Studi Teknik Komputer, Fakultas Teknik, Universitas Indonesia

DEWAN PENGUJI

Pembimbing : Muhammad Salman ST, MIT ( )

Penguji : Ir. Endang Sriningsih, MT, Si ( )

Penguji : Dr. Ir.Anak Agung Putri Ratna M.Eng. ( )

Ditetapkan di : Depok

Tanggal : 03 Januari 2011

LEMBAR PERSETUJUAN

Skripsi dengan judul:

Simulasi Dan Analisa Pengaruh *TCP Windowing* Pada *Transport Layer* Terhadap Peningkatan Kinerja Jaringan Berbasis *Virtual Private Network* (VPN) Menggunakan Simulator OPNET

dibuat untuk melengkapi sebagian persyaratan menjadi Sarjana Teknik pada Program Studi Teknik Komputer, Departemen Teknik Elektro Universitas Indonesia dan disetujui untuk diajukan dalam presentasi skripsi

Depok, 17 Desember 2010

Dosen Pembimbing,

Muhammad Salman, ST, MIT

NIP : 196903291997031002

KATA PENGANTAR

Alhamdulillah, segala puji dan syukur saya panjatkan kepada Tuhan Yang Maha Esa, karena atas berkat dan rahmat-Nya saya dapat menyelesaikan Tugas akhir ini. Penulisan Tugas akhir ini dilakukan dalam rangka memenuhi salah satu syarat untuk mencapai gelar Sarjana Teknik Jurusan Teknik Komputer pada Fakultas Teknik Universitas Indonesia. Saya menyadari bahwa, tanpa bantuan dan bimbingan dari berbagai pihak, dari masa perkuliahan sampai pada penyusunan tugas akhir ini, sangatlah sulit bagi saya untuk menyelesaikan tugas akhir ini. Oleh karena itu, saya mengucapkan terima kasih kepada:

1. Muhammad Salman, ST, MIT, selaku dosen pembimbing yang telah membimbing saya selama satu setengah tahun ini. Setiap pertemuan dengan Beliau selalu menimbulkan inspirasi baru.
2. Kedua Orang tua dan keluarga saya yang telah memberikan bantuan dukungan material dan moral.
3. Sahabat-sahabat saya, Eko atas bantuannya yang begitu banyak selama pengerjaan skripsi ini, dan teman-teman seperjuangan Winda, Monik, Barnas, Cesil, Ramdhan, Dedi untuk suntikan semangat tiada henti, yang telah banyak membantu saya dalam menyelesaikan tugas akhir ini.

Akhir kata, saya berharap Tuhan Yang Maha Esa berkenan membalas segala kebaikan semua pihak yang telah membantu. Semoga tugas akhir ini membawa manfaat bagi pengembangan ilmu.

Depok, Desember 2010

Penulis

HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS

Sebagai sivitas akademik Universitas Indonesia, saya yang bertanda tangan di bawah ini:

Nama : Yomma Hendra Putra
NPM : 0606078323
Program Studi : Teknik Komputer
Departemen : Teknik Elektro
Fakultas : Teknik
Jenis karya : Skripsi

demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Indonesia **Hak Bebas Royalti Noneksklusif** *(Non-exclusive Royalty-Free Right)* atas karya ilmiah saya yang berjudul :

Simulasi Dan Analisa Pengaruh *TCP Windowing* Pada *Transport Layer* Terhadap Peningkatan Kinerja Jaringan Berbasis *Virtual Private Network* (VPN) Menggunakan Simulator OPNET

beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Noneksklusif ini Universitas Indonesia berhak menyimpan, mengalihmedia / formatkan, mengelola dalam bentuk pangkalan data *(database)*, merawat, dan memublikasikan tugas akhir saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Depok
Pada tanggal : 17 Desember 2010
Yang menyatakan

(Yomma Hendra Putra)

ABSTRAK

Nama : Yomma Hendra Putra

Program Studi : Teknik Komputer

Judul : Simulasi Dan Analisa Pengaruh *TCP Windowing* Pada *Transport Layer* Terhadap Peningkatan Kinerja Jaringan Berbasis *Virtual Private Network* (VPN) Menggunakan Simulator OPNET

Virtual Private Network (VPN) hadir untuk menjawab permasalahan keamanan yang kerap kali muncul pada transmisi data melalui jaringan publik berskala besar seperti *Wide Area Network* (WAN). Teknologi VPN menggunakan metode enkripsi-dekripsi untuk melindungi data yang dikirim melalui jaringann publik, Namun sayangnya keunggulan kemanan yang ditawarkan VPN ini harus dibayar dengan peningkatan delay pada jaringan. Penelitian ini mencoba untuk mengajukan solusi yang dapat diimplementasikan untuk mengurangi delay pada VPN. Karena tujuannya adalah mengurangi delay, maka aplikasi yang dijalankan pada VPN adalah aplikasi yang sensitif terhadap delay, dan pada penelitian ini digunakan *Streaming Multimedia* yang berjalan pada aplikasi HTTP.

Dalam proses pengujian dilakukan empat skenario untuk melihat performa jaringan, parameter yang dilihat yaitu packet loss, delay, throughput dan page response time dari aplikasi HTTP. Skenario pertama melihat performa jaringan pada kondisi normal, skenario kedua menambahkan saluran VPN pada jaringan, skenario ketiga merubah ukuran TCP Window dari 8k menjadi 32k dan skenario keempat hanya digunakan untuk perbandingan, yaitu menguji seberapa besar peninkata performa jaringan dengan meng-upgrade link WAN dari DS1 ke DS3. Hasil penelitian menunjukkan bahwa dengan menaikkan ukuran TCP window menjadi 32k dapat mengurangi delay TCP sebesar 0.02s, page response time berkurang sebesar 0.1s, queuing delay berkurang sebesar 0.2ms.

Kata kunci: VPN, WAN, Streaming Multimedia, Delay, Packet Loss, Throughput, TCP Window Size, dan TCP/IP.

ABSTRACT

Name : Yomma Hendra Putra

Major : Computer Engineering

Title : Simulation and Analysis of The Impact of TCP Windowing Transport Layer on Performance Improvement of VPN-Based Network Using OPNET Simulator

Virtual Private Network (VPN) comes out as a solution addressed to the security issues that emerge in Wide Area Network (WAN). VPN technology uses encryption-decryption method to secure the data transferred through public network. Unfortunately, this advantage of security must be paid with the lack of network performance due to the delay increment. The aim of this research is to propose the solution to reduce the delay on VPN network. Streaming Multimedia application was chosen to assess the network performance because of its sensitivity of delay. The streaming multimedia ran over the HTTP application.

VPN network performance was examined by conducting four scenarios, and there are several network parameters that measured during the simulation such as delay, throughput, HTTP page response time, and packet loss. The first scenario tried to assess the network performance on the normal condition, second scenario assessed the network performance with VPN implementation, third scenario assessed the performance of VPN-based network with higher TCP window size of 32k, as comparison the fourth scenario tried to see the improvement of the VPN network by upgrading the WAN link from DS1 to DS3. The result shows that by increasing the TCP window size would reduce the delay up to approximately 0.02s, and page response time reduced up to approximately 0.1s, and queuing delay on WAN link reduced approximately 0.2ms.

Keywords: VPN, WAN Streaming Multimedia, Delay, Packet Loss, Throughput, TCP Window Size, and TCP/IP

DAFTAR ISI

DAFTAR GAMBAR

DAFTAR TABEL

BAB 1
PENDAHULUAN

1.1 Latar Belakang

Akses terhadap informasi telah menjadi kebutuhan mendasar di era teknologi informasi ini, akses informasi bukan hanya menjadi hal yang vital bagi perusahaan dan organisasi, bahkan individu pun memiliki kebutuhan yang besar akan akses informasi. Teknologi jaringan komputer diciptakan untuk memungkinkan komputer-komputer yang berbeda dapat saling berkomunikasi dan bertukar data, teknologi ini kemudian berkembang menjadi jaringan-jaringan yang lebih banyak dan lebih besar sesuai dengan tuntutan kebutuhan banyak pihak. Sekarang kita dapat menyaksikan interkoneksi yang lebih luas antara komputer-komputer dari berbagai belahan dunia, komputer ini terhubung melalui sebuah jaringan sangat besar yang terbentuk dari jaringan-jaringan kecil yang saling terhubung dengan saluran data berkecepatan tinggi, jaringan dalam skala besar ini disebut dengan *Wide Area Network* (WAN).

Dengan adanya WAN memungkinkan setiap individu untuk mendapatkan akses informasi dari komputer manapun yang terhubung dengan jaringan ini, akibatnya tersedia begitu banyak informasi yang beragam dan semua orang dapat mengaksesnya dengan mudah. Namun hal ini kemudian juga menjadi permasalahan ketika komunikasi antar dua titik yang melalui WAN harus bersifat rahasia, beberapa informasi yang ditransmisikan melalui jaringan terkadang memang bersifat rahasia karena berisi data-data penting internal suatu organisasi sehingga data ini tidak boleh jatuh ketangan pihak ketiga. Jika menyangkut masalah keamanan, WAN menunjukkan sisi kelemahannya karena pada WAN semua interkoneksi memungkinkan siapapun untuk mendapatkan apapun yang beredar didalam jaringan, bahkan data-data penting yang sifatnya rahasia pun bisa didapatkan. Oleh karena itu perlu ada suatu teknologi yang menjamin keamanan data yang ditransfer melalui jaringan publik ini.

Virtual Private Network (VPN) hadir sebagai salah satu solusi untuk mengamankan data yang ditransfer melalui WAN, teknologi ini memungkinkan data yang dikirim dibuat dalam bentuk ter-enkripsi dan hanya bisa dibaca ketika

sudah di-dekripsikan kembali sehingga tidak bisa dengan mudah dikuasai oleh pihak ketiga. Namun ternyata proses enkripsi dan dekripsi pada VPN membuat delay didalam jaringan bertambah karena proses ini juga membutuhkan waktu. Pada akhirnya kemanan data pada VPN harus dibayar dengan penambahan delay pada jaringan. Skripsi ini mencoba mengajukan salah satu solusi untuk mengurangi delay yang terjadi pada jaringan dengan tunel VPN, solusi pertama yaitu dengan merubah ukuran TCP Window menjadi lebih besar, akibatnya data yang ditransfer dalam sekali transmisi akan lebih besar sehingga waktu transmisi yang diperlukan untuk mengirim keseluruhan data menjadi lebih cepat. Solusi kedua mencoba melihat perubahan performa jaringan jika link WAN di-upgrade menjadi link dengan kecepatan yang lebih baik.

1.2 Perumusan Masalah

Penelitian dalam skripsi ini bertujuan untuk melihat efek perubahan ukuran TCP Window terhadap perubahan delay pada VPN dan performa aplikasi yang berjalan pada jaringan dengan VPN menggunakan beberapa skenario jaringan yang berbeda. Pengujian dilakukan dengan melihat hasil dari beberapa solusi untuk mengurangi delay, solusi pertama adalah dengan memperbesar ukuran TCP Window dan sebagai perbandingan dicoba juga solusi kedua yaitu dengan mengganti link WAN menjadi saluran dengan kecepatan yang lebih tinggi. Analisa dilakukan untuk melihat perubahan performa jaringan VPN dan aplikasi yang berjalan didalamnya, untuk kemudian melihat solusi mana yang kira-kira paling tepat untuk diterapkan. Penelitian ini dilakukan melalui simulasi menggunakan software simulator OPNET IT Guru Academic Edition 9.1.

1.3 Tujuan Penulisan

Penulisan skripsi ini bertujuan untuk melihat efek perubahan ukuran TCP window terhadap performa jaringan berbasis VPN. Selain itu juga untuk menguji sejauh mana keefektifan penyesuaian ukuran TCP window untuk meningkatkan performa jaringan dibandingkan dengan performa jaringan yang didapat jika menggunakan infrastruktur yang lebih baik yaitu dengan meng-upgrade link WAN dari DS1 ke DS3.

1.4 Batasan Masalah

Dalam skripsi ini terdapat beberapa pembatasan masalah, diantaranya yaitu :

1. Pengujian dilakukan melalui simulasi menggunakan software simulator OPNET IT Guru Academic Edition 9.1.
2. Pengukuran parameter dilakukan pada link WAN dan aplikasi yang berjalan pada VPN.
3. Karena parameternya adalah delay, maka aplikasi yang dijalankan adalah aplikasi yang sensitif terhadap delay, yaitu aplikasi multimedia *streaming* yang berjalan melalui aplikasi HTTP.
4. Analisa terutama tertuju pada pengaruh perubahan TCP window terhadap delay VPN sebagai solusi untuk mengurangi delay, dan untuk perbandingan dilihat pilihan solusi lain dengan meng-upgrade link pada WAN.

1.5 Sistematika Penulisan

Sistematika penulisan skripsi ini adalah sebagai berikut:

Bab 1 : Pendahuluan

Membahas tentang latar belakang penulisan, perumusan masalah, tujuan penulisan, batasan masalah serta sistematika penulisan.

Bab 2 : Teknologi WAN, Protokol TCP/IP, VPN, QoS Jaringan, dan Simulator OPNET.

Penjelasan tentang dasar teori yang berkaitan dengan konsep jaringan berskala besar (WAN), transmisi data pada jaringan dengan protokol TCP/IP, Virtual Private Network mencakup teknologi tunneling pada VPN dan protokol-protokol VPN, Parameter performa (QoS) jaringan komputer, dan software simulator OPNET IT Guru Academic Edition 9.1.

Bab 3 : Simulasi VPN pada WAN dan Metode Pengambilan Data

Bab ini menjelaskan topologi jaringan yang akan digunakan. Pembahasan meliputi konstruksi dan konfigurasi jaringan pada software simulator OPNET dan beberapa skenario pengambilan data.

Bab 4 : Pengambilan Data dan Analisa

Pada bab ini akan dibahas analisa data untuk mengetahui performansi jaringan VPN saat aplikasi streaming multimedia dijalankan melalui HTTP, dengan perubahan ukuran TCP Window, dan perubahan link WAN.

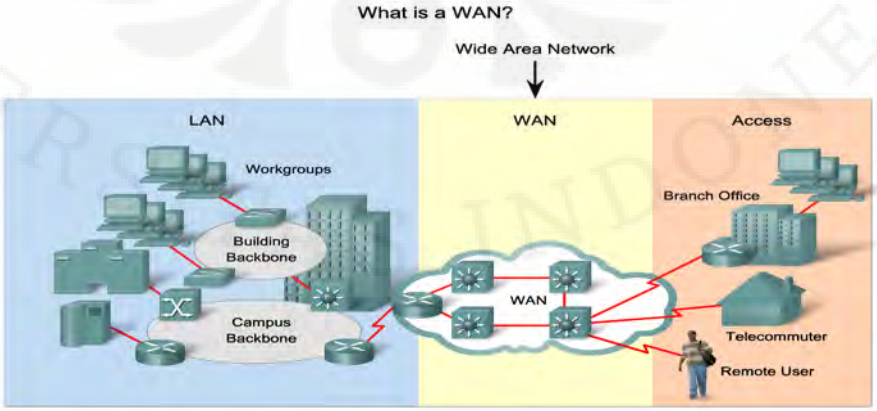
Bab 5 : Kesimpulan

Bab ini berisi kesimpulan dan saran dari penulisan skripsi ini.

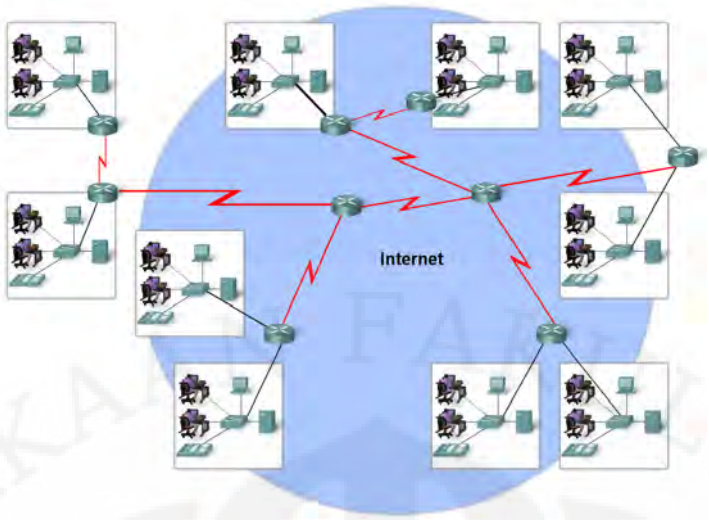
BAB 2
TEKNOLOGI WAN, VPN, PROTOKOL TPC/IP, QoS JARINGAN KOMPUTER, DAN SIMULATOR OPNET

2.1 Teknologi *Wide Area Network* (WAN)

Performa jaringan komputer memiliki keterbatasan terhadap ukuran dan jarak, semakin banyak jumlah komputer dalam satu jaringan dan semakin jauh jarak antar perangkat didalam jaringan akan semakin menurunkan performa jaringan tersebut. Namun dengan teknologi yang ada saat ini, sangat memungkinkan untuk membuat jaringan komputer yang lebih luas yang bahkan dapat mencakup seluruh permukaan bumi dengan tetap menjaga agar kualitas komunikasi antar jaringan tersebut tetap baik sehingga menciptakan komunikasi tanpa batas, yaitu dengan menghubungkan jaringan-jaringan kecil (LAN) dengan sambungan (*link*) berkecepatan tinggi sehinga jaringan-jaringan ini kemudian membentuk jaringan yang lebih besar yang disebut dengan *Wide Area Network* (WAN). WAN dapat didefinisikan sebagai sekumpulan jaringan-jaringan komputer yang saling terhubung membentuk sebuah jaringan besar yang mencakup area geografis yang lebih luas. WAN biasanya memanfaatkan fasilitas transmisi data yang sudah ada berupa jalur layanan umum yang biasanya dimiliki oleh perusahaan-perusahaan telekomunikasi. Terdapat perbedaan karateristik antara WAN dengan LAN, tidak seperti LAN yang umumnya identik dengan perangkat-perangkat seperti PC, *shared printer*, switch/hub, dan server lokal. WAN lebih identik dengan router, modem, dan link berkecepatan tinggi seperti T1, T3, dan lainnya.

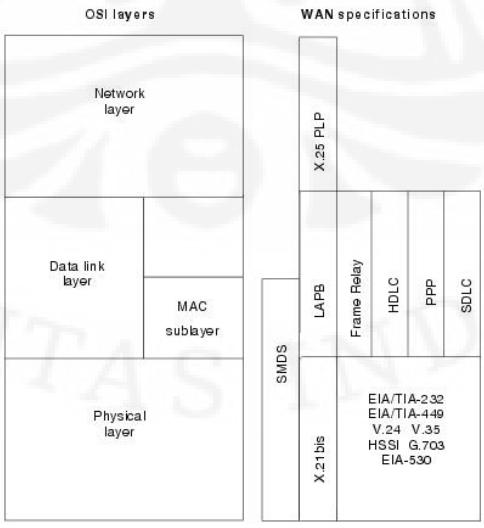


Gambar 2.1 Bentuk jaringan Wide Area Network (WAN) [1]



Gambar 2.2 Jaringan WAN yang lebih luas membentuk internet [1]

Teknologi WAN beroperasi pada tiga lapisan terbawah model referensi OSI, yaitu *physical layer*, *data link layer*, dan *network layer*. Standar akses WAN secara spesifik menjelaskan metode pengiriman di *physical layer* dan kebutuhan di *data link layer*, termasuk pengalamatan fisik, aliran data dan enkapsulasi[2]. Standarisasi untuk teknologi WAN ditetapkan beberapa lembaga resmi internasional diantaranya adalah Internet Engineering Task Force (IETF), International Organization for Standardization (ISO), International Telecommunication Union-Telecommunication Standardization Sector (ITU-T), Consultative Committee for International Telegraph and Telephone (CCITT), dan Electronics Industries Association (EIA).

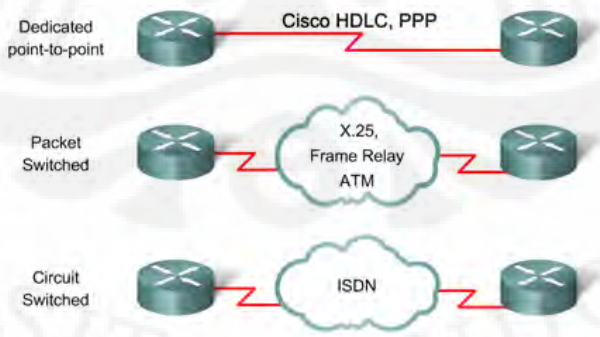


Gambar 2.3 WAN beroperasi pada tiga lapisan terbawah OSI layer [2]

Terdapat cukup banyak teknologi dan protokol yang digunakan dalam WAN, namun secara umum ada tiga teknologi WAN yang masing-masingnya memiliki contoh sendiri yang akan dijelaskan lebih lanjut pada bagian berikutnya. Tabel dibawah ini memperlihatkan pilihan teknologi WAN sebagai berikut:

Tabel 2.1 Teknologi saluran dan protokol pada WAN [3]

Opsi Teknologi	**Penjelasan**	**Kelebihan**	**Kekurangan**	**Contoh Protokol**
Leased Line	Koneksi *point-to-point* antara dua komputer atau network (LAN)	Lebih aman	Harga Mahal	PPP, HDLC, SDLC, HNAS
Circuit Switching	Jalur khusus yang dibentuk ketika dibutuhkan saat transfer data antara dua titik, contohnya adalah koneksi *Dial-Up*	Lebih murah	Dibutuhkan pengaturan saluran panggilan	ISDN
Packet Switching	Perangkat mengirim paket melalui jalur point-to-point atau point-to-multiple yang terbentuk disepanjang jaringan pembawa (*carrier network*). Paket ditansmisikan melalui *Permanent Virtual Circuit* (PVC) atau *Switched Virtual Circuit* (SVC)	Lebih murah	Harus berbagi media saluran yang sama	X.25, Frame Relay
Cell Relay	Sebenarnya sama dengan packet switching, namun teknologi ini menggunakan sel dengan panjang tetap (*fixed length cells*) bukan *variable length packet*.	Sangat bagus untuk penggunaan simultan data dan suara (*voice*)	overhead	ATM



Gambar 2.4 Jenis-jenis Teknologi dan Protokol WAN [4]

a. **Point-to-Point Links,** saluran point-to-point menyediakan jalur komunikasi khusus yang di alokasikan untuk komunikasi dari *client* ke jaringan remote melalui jaringan pembawa (*carrier network*) seperti

jaringan milik perusahaan telekomunikasi. Saluran ini biasanya disewa dari perusahaan telekomunikasi oleh karena itu sering disebut juga sebagai *Leased Line* (saluran yang disewa) agar secara khusus dialokasikan untuk transmisi data dari *client* ke *remote network*.

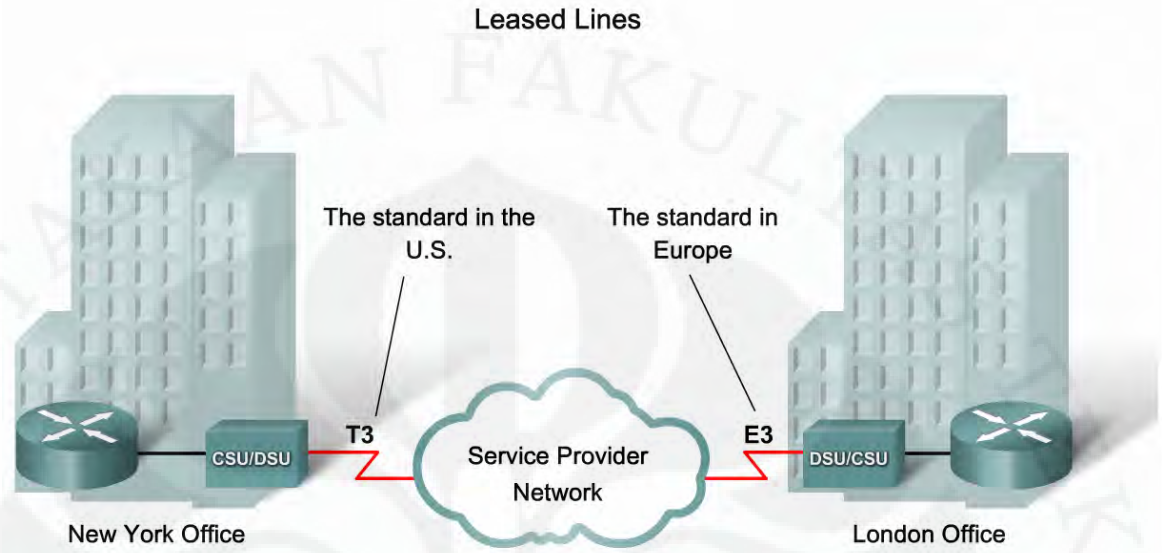
b. **Circuit Switching,** teknologi ini hadir untuk menjawab persoalan yang muncul pada teknologi point-to-point, yaitu biaya yang sangat mahal. Alih-alih menyewa saluran secara ekslusif kepada perusahaan telekomunikasi, *circuit switching* memanfaatkan jaringan publik (*shared network*) untuk berkomunikasi. Circuit switching memungkinkan terbentuknya saluran sementara jika dibutuhkan untuk transmisi data antara dua titik, dan lalu memutuskan saluran tersebut ketika tidak digunakan. Cara kerjanya mirip dengan saluran telepon, contohnya adalah *Integrated Service Digital Network* (ISDN).

c. **Packet Switching,** teknologi ini sama hal nya dengan circuit switching, yaitu lebih murah dibandingkan dengan *leased line*. Pada packet switching koneksi terjadi antara beberapa titik (*multiple points*) dalam jaringan, teknologi ini memungkinkan penggunanya untuk membagi sumber daya dalam jaringan. Paket yang dikirim dipecah-pecah kedalam bentuk paket yang lebih kecil untuk dikirim melalui saluran virtual pada jaringan pembawa (*carrier network*). Contoh packet switching adalah Asychronous Transfer Mode (ATM), Frame Relay, Switched Mulimegabit Data Service (SMDS), dan X.25 [2].

2.1.1 Leased Line

Leased Line merupakan bagian dari *point-to-point communication*, dimana konesi jaringan dibangun dengan sebuah saluran khusus yang disewa dari perusahaan telekomunikasi agar dapat dialokasikan untuk transmisi data dari client ke remote network, leased line dapat berjalan pada saluran analog (seperti saluran telepon) maupun digital. Biasanya saluran seperti ini sering digunakan oleh perusahaan yang mengutamakan kualitas komunikasi premium, karena leased line merupakan saluran yang sifatnya privat dan tidak seperti dial-up, leased line selalu aktif. Teknologi ini menawarkan tingkat keamanan yang sangat

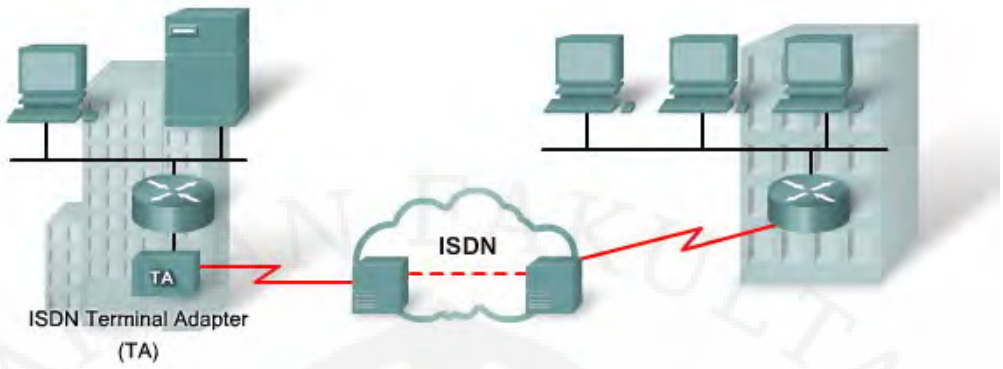
baik, namun sayangnya kelebihan ini harus dikompensasi dengan biaya yang mahal. Besarnya biaya ditentukan dari besarnya *bandwidth* dan jarak antar node yang terhubung dalam saluran tersebut.



Gambar 2.5 Saluran Leased Line [5]

2.1.2 Intergrated Service Digital Network (ISDN)

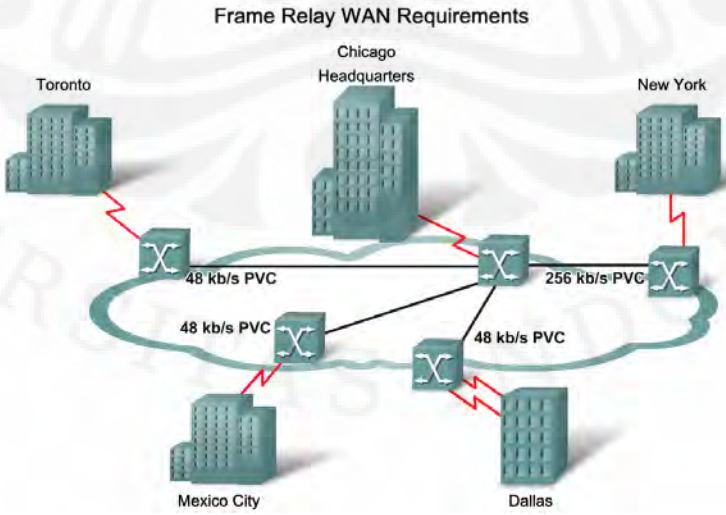
ISDN merupakan revolusi dari jaringan telepon konvensional yaitu *Public Switched Telephone Network* (PSTN) yang dirombak menjadi jaringan digital terintegrasi sehingga dapat melayani komunikasi data, suara, dan bahkan video. ISDN muncul sebagai sarana telekomunikasi yang menawarkan fleksibilitas tinggi namun dengan biaya yang rendah. Keuntungan ISDN antara lain kecepatan dan kualitas yang baik dalam pengiriman data bahkan 10 kali lebih cepat dibandingkan PSTN, efisien karena hanya perlu satu saluran saja untuk mengirim berbagai jenis layanan (data, suara, video), fleksibel dengan *single interface* untuk terminal bervariasi, dan hemat biaya karena hanya membutuhkan satu terminal tunggal untuk audio dan video. Ada dua jenis pelayanan ISDN yaitu *Basic Rate Interface* (BRI) dan *Primary Rate Interface* (PRI).



Gambar 2.6 Integrated Service Digital Network (ISDN) [5]

2.1.3 Frame Relay

Sebagai salah satu contoh dari teknologi packet switching, frame relay memungkinkan dua titik yang saling berkomunikasi untuk berbagi sumber daya dalam jaringan, termasuk medium jaringan dan bandwidth yang tersedia. Seperti halnya packet switching, frame relay menggunakan teknik *variable-length packet* yang meningkatkan efisiensi dan fleksibilitas dalam pengiriman paket, data dialirkan ke berbagai segmen didalam jaringan hingga sampai ke tujuan. Teknik *statistical multiplexing* juga digunakan untuk mengontrol akses jaringan pada *packet-switched network*, keuntungannya adalah dapat mengefisiensi dan lebih fleksibel dalam penggunaan bandwidth.



Gambar 2.7 Infrastruktur Jaringan Frame Relay [6]

Frame pada frame relay dikirimkan melalui sirkuit virtual yang dapat berupa *permanent virtual circuit* (PVC) atau *switched virtual circuit* (SVC). PVC adalah koneksi yang terbentuk untuk menghubungkan dua peralatan secara terus menerus tanpa memperhitungkan apakah sedang ada komunikasi data yang terjadi di dalam sirkuit tersebut. PVC tidak memerlukan proses pembangunan panggilan seperti pada SVC dan memiliki 2 status kerja:

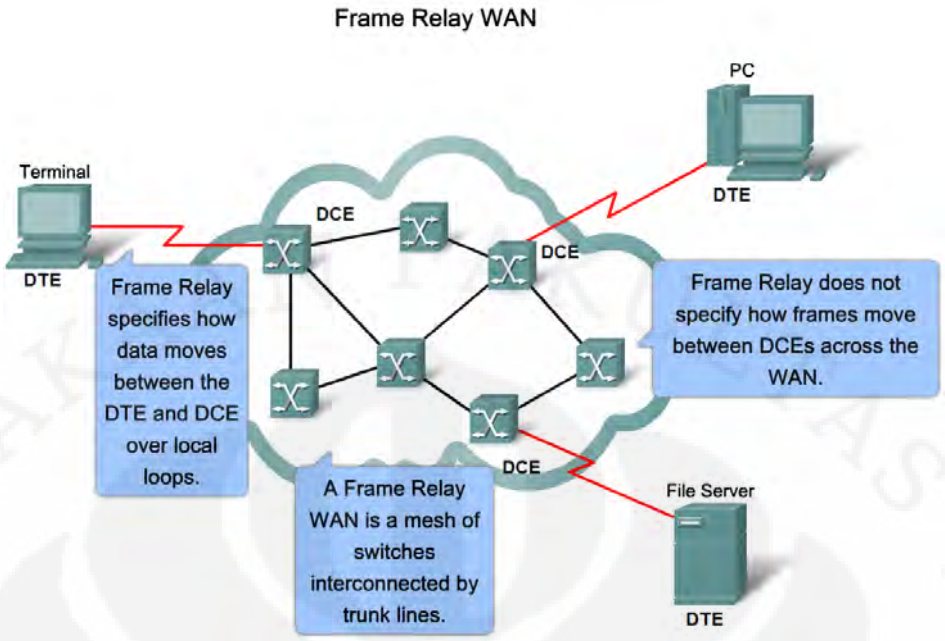
a. *Data Transfer*, pengiriman data sedang terjadi dalam sirkuit.
b. *Idle*, koneksi antar titik masih aktif tapi tidak ada data yang dikirimkan dalam sirkuit.

SVC adalah koneksi sementara yang terbentuk hanya pada kondisi dimana pengiriman data berlangsung. Status-status dalam koneksi ini adalah:

a. *Call Setup*, hubungan antar perangkat sedang dibangun.
b. *Data Transfer*, data dikirimkan antar perangkat dalam sirkuit virtual yang telah dibangun.
c. *Idle*, ada koneksi aktif yang telah terbentuk, tetapi tidak ada data yang lewat di dalamnya.
d. *Call Termination*, pemutusan hubungan antar perangkat, terjadi saat waktu idle melebihi patokan yang ditentukan.

Terdapat dua perangkat yang harus ada dalam frame relay yaitu *Data Terminal Equipment* (DTE) dan *Data Circuit-terminating Equipment* (DCE). DTE biasanya merupakan perangkat-perangkat yang berada pada sisi *client/customer*, seperti PC, router, dan bridge, sementara DCE adalah perangkat-perangkat *internetworking* yang ada didalam jaringan pembawa yang berfungsi untuk menyediakan pelayanan *clocking* dan *switching* didalam jaringan.

Frame relay akan menentukan bagaimana data mengalir dari DTE ke DCE, namun tidak menentukan bagaimana data akan bergerak diantara perangkat-perangkat DCE yang tersebar didalam jaringan WAN, oleh karena itu frame relay biasanya akan menghasilkan delay.



Gambar 2.8 DTE dan DCE pada Frame Relay [6]

Frame relay sering dianggap sebagai pengganti teknologi X.25 namun dengan beberapa kemampuan yang dikurangi diantaranya *windowing* dan kemampuan untuk mentransmisi ulang data (*retransmission*). Hal ini dikarenakan frame relay beroperasi pada WAN dengan pelayanan koneksi yang lebih handal dan lebih dapat dipercaya jika dibandingkan dengan jaringan yang ada pada sekitar akhir tahun 1970an sampai awal 1980an dimana ketika itu teknologi X.25 sangat populer. Frame relay yang hanya beroperasi pada layer 2 menawarkan performa yang lebih tinggi dan efisiensi transmisi yang lebih baik dibanding X.25 yang beroperasi pada layer 2 dan layer 3. Hal ini membuat frame relay menjadi pilihan yang lebih pas ketimbang X.25 untuk kondisi saat ini [2].

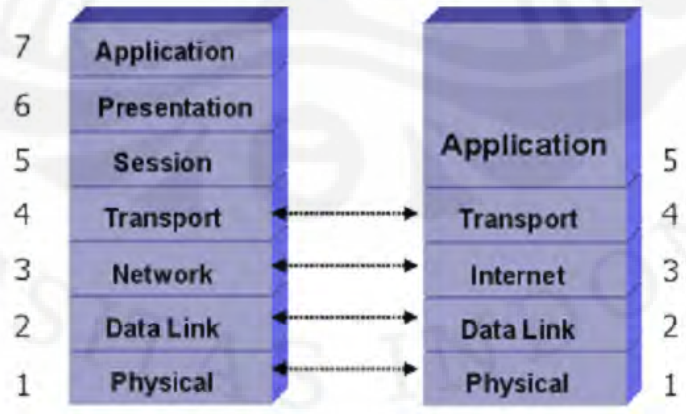
2.2 Protokol TCP/IP

TCP/IP merupakan sebuah protokol yang disepakati untuk mengatur proses komunikasi yang terjadi didalam jaringan komputer. Dalam sebuah jaringan, perangkat yang saling berkomunikasi bisa jadi merupakan perangkat-perangkat dengan jenis yang berbeda secara *hardware* atau *software* atau bahkan keduanya. TCP/IP hadir sebagai protokol yang menjembatani perbedaan-perbedaan perangkat didalam jaringan, protokol ini merupakan seperangkat aturan

yang disepakati tentang bagaimana proses komunikasi didalam jaringan berlangsung. TCP/IP digunakan sebagai sarana pengirim data atau kendali melalui jaringan komputer.

Sejarah TCP/IP dimulainya dari lahirnya ARPANET yaitu jaringan paket switching digital yang didanai oleh DARPA (Defence Advanced Research Projects Agency) pada tahun 1969. Sementara itu ARPANET terus bertambah besar sehingga protokol yang digunakan pada waktu itu tidak mampu lagi menampung jumlah node yang semakin banyak. Oleh karena itu DARPA mendanai pembuatan protokol komunikasi yang lebih umum, yakni TCP/IP yang kemudian diadopsi menjadi standard ARPANET pada tahun 1983. Untuk memudahkan proses konversi, DARPA juga mendanai suatu proyek yang mengimplementasikan protokol ini ke dalam BSD UNIX, sehingga dimulailah perkawinan antara UNIX dan TCP/IP. Pada awalnya internet digunakan untuk menunjukan jaringan yang menggunakan internet protocol (IP) tapi dengan semakin berkembangnya jaringan, istilah ini sekarang sudah berupa istilah generik yang digunakan untuk semua kelas jaringan. Internet digunakan untuk menunjuk pada komunitas jaringan komputer worldwide yang saling dihubungkan dengan protokol TCP/IP [7].

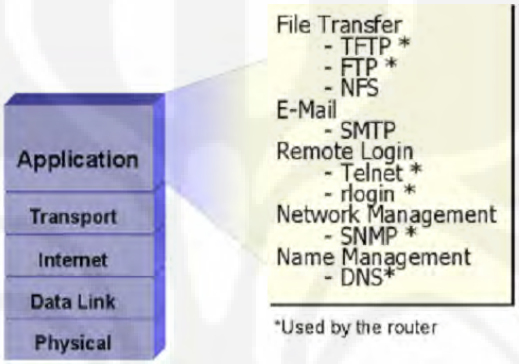
Susunan protokol pada TCP/IP mirip dengan OSI layer, hanya saja pada TCP/IP hanya ada 5 layer sementara OSI memiliki 7 layer.



Gambar 2.9 Perbandingan layer pada OSI dan TCP/IP [8]

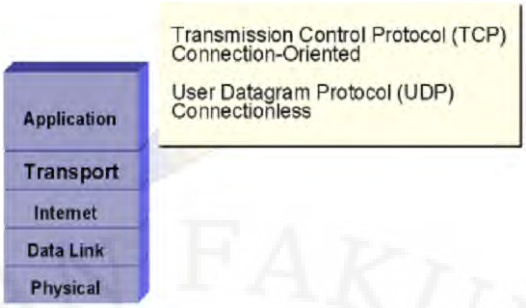
Informasi TCP/IP ditransfer dalam sebuah urutan *datagram*. Setiap pesan yang ditransmisikan akan dipecah dan dikirim berupa rentetan datagram yang kemudian disisi penerima akan disusun kembali seperti pesan awal. Setiap lapisan dalam TCP/IP memiliki fungsi masing-masing, yaitu :

a. *Application Layer*, protokol Application terdiri dari *file transfer, email, remote login,* dan manajemen jaringan [8]. Protokol yang beroperasi pada layer ini mencakup; DNS, TFTP, TLS/SSL, FTP, Gopher, HTTP, IMAP, IRC, NNTP, POP3, SIP, SMTP, SMPP, SNMP, SSH,Telnet, Echo, RTP, PNRP, rlogin, dan ENRP.



Gambar 2.10 Appication layer pada TCP/IP [8]

b. *Transport Layer*, mendefinisikan cara-cara untuk melakukan pengiriman data antara end to end host. Lapisan ini menjamin bahwa informasi yang diterima pada sisi penerima adalah sama dengan informasi yang dikirimkan pada pengirim. Untuk itu, lapisan ini memiliki beberapa fungsi penting antara lain Flow Control, pengiriman data yang telah dipecah menjadi paket-paket tersebut harus diatur sedemikian rupa agar pengirim tidak sampai mengirimkan data dengan kecepatan yang melebihi kemampuan penerima dalam menerima data. Error Detection, pengirim dan penerima juga melengkapi data dengan sejumlah informasi yang bisa digunakan untuk memeriksa data yang dikirimkan bebas dari kesalahan. Jika ditemukan kesalahan pada paket data yang diterima, maka penerima tidak akan menerima data tersebut. Pengirim akan mengirim ulang paket data yang mengandung kesalahan tadi. Namun hal ini dapat menimbulkan delay yang cukup berarti. Protokol yang beroperasi pada layer ini adalah TCP, UDP, DCCP, SCTP, IL, RUDP, dan RSVP [7].



Gambar 2.11 Transport layer pada TCP/IP

c. *Internet Layer*, lapisan ini berpadanan dengan lapisan *Network* pada OSI, tuganya adalah untuk mendefinisikan bagaimana hubungan dapat terjadi antara dua perangkat yang berada dalam jaringan yang berbeda. Pada jaringan Internet yang terdiri atas puluhan juta host dan ratusan ribu jaringan lokal, lapisan ini bertugas untuk menjamin agar suatu paket yang dikirimkan dapat menemukan tujuannya dimana pun berada. Oleh karena itu, lapisan ini memiliki peranan penting terutama dalam mewujudkan internetworking yang meliputi wilayah luas (*Worldwide Internet*). Fungsi yang dijalankan oleh Internet layer adalah *Addresing* dan *Routing*. Addressing, yakni melengkapi setiap datagram dengan alamat Internet dari tujuan. Alamat pada protokol inilah yang dikenal dengan Internet Protocol Address (IP Address). Karena pengalamatan (addressing) pada jaringan TCP/IP berada pada level ini (software), maka jaringan TCP/IP independen dari jenis media dan komputer yang digunakan. Fungsi routing menentukan ke mana datagram akan dikirim agar mencapai tujuan yang diinginkan. Fungsi ini merupakan fungsi terpenting dari Internet Protocol (IP). Sebagai protokol yang bersifat connectionless, proses routing sepenuhnya ditentukan oleh jaringan. Pengirim tidak memiliki kendali terhadap paket yang dikirimkannya untuk bisa mencapai tujuan. Router-router pada jaringan TCP/IP lah yang sangat menentukan dalam penyampaian datagram dari penerima ke tujuan. Protokol pada layer ini adalah IP (IPv4, & IPv6), ICMP, IGMP, dan ICMPv6 [7].

Gambar 2.12 Internet layer pada TCP/IP

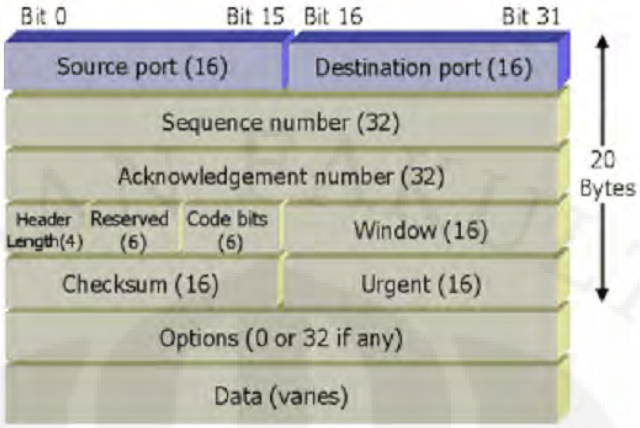
d. *Data Link Layer,* lapisan ini mengatur penyaluran data frame-frame data pada media fisik yang digunakan secara handal. Lapisan ini biasanya memberikan servis untuk deteksi dan koreksi kesalahan dari data yang ditransmisikan. Beberapa contoh protokol yang digunakan pada lapisan ini adalah X.25 jaringan publik, Ethernet untuk jaringan Etehernet, AX.25 untuk jaringan Paket Radio dsb.
e. *Physical Layer*, merupakan lapisan terbawah yang mendefinisikan besaran fisik seperti media komunikasi, tegangan, arus, dsb. Lapisan ini dapat bervariasi bergantung pada media komunikasi pada jaringan yang bersangkutan.TCP/IP bersifat fleksibel sehingga dapat mengintegrasikan berbagai jaringan dengan media fisik yang berbeda-beda [7].

2.2.1 Transmision Control Protocol (TCP)

TCP merupakan pengarah koneksi, sebuah protokol yang berada pada layer transport di TCP/IP *protocol stack*. Fungsi umum protokol TCP adalah:

1. TCP bertugas memecah pesan-pesan menjadi beberapa segmen, menyatukan kembali (*reassemble*) ketika sampai ditujuan, mengirimkan kembali apapun yang tidak diterima, dan menyatukan kembali pesan-pesan tersebut dari beberapa segmen.
2. TCP menyediakan sirkuit virtual antara aplikasi-aplikasi pada *end-user*.

Gambar berikut ini adalah anatomi sebuah segmen yang terdapat pada protokol TCP:



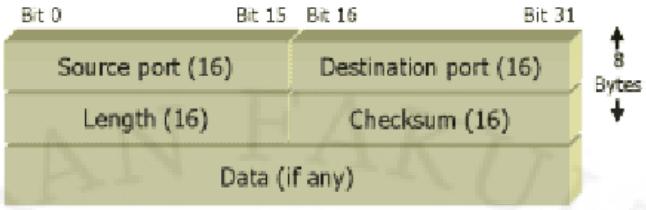
Gambar 2.13 Segment pada TCP [8]

Source Port, nomor pengirim data (16bit). ***Destination Port***, nomor tujuan (16bit). ***Sequence Number***, nomor yang digunakan untuk memastikan kedatangan urutan data yang benar (32bit). ***Acknowledgement Number***, sinyal untuk memberitahu data yang telah diterima dan perkiraan TCP octet berikutnya (32bit). ***Header Length***, nomor yang terdiri 32-bit yang menandakan header suatu segmen (32bit). ***Reserved***, diset menjadi nol (6bit). ***Code Bits***, fungsi pengendali termasuk setup dan akhiran sesi (6bit). ***Window***, nomor octet yang dapat diterima oleh suatu perangkat (16bit). ***Checksum***, perhitungan checksum dari suatu header dan data field (16bit). ***Urgent***, menunjukkan akhir dati sebuah data yang mendesak (16bit). ***Option***, suatu penegasa saat ini, ukuran segmen TCP maksismum (0 atau 32bit jika ada). ***Data***, layer atas data protocol (panjang bervariasi).

2.2.2 User Datagram Protocol (UDP)

UDP merupakan protokol yang didesain untuk aplikasi-aplikasi yang menyediakan "*error recovery process*" sendiri. UDP merupakan bentuk protokol yang sederhana dan efisien namun tidak dapat diandalkan, protokol ini bersifat *connectionless* dan tanpa acknowledgement selain itu juga bergantung pada protokol layer diatasnya untuk tingkat kehandalan. Meskipun UDP bertugas mengirimkan pesan-pesan, tidak software yang dapat memeriksa pengiriman

segmen pesan tersebut pada layer ini. Paket UDP terbagi dalam segmen-segmen dengan panjang header yang selalu sama yaitu 64 bit.

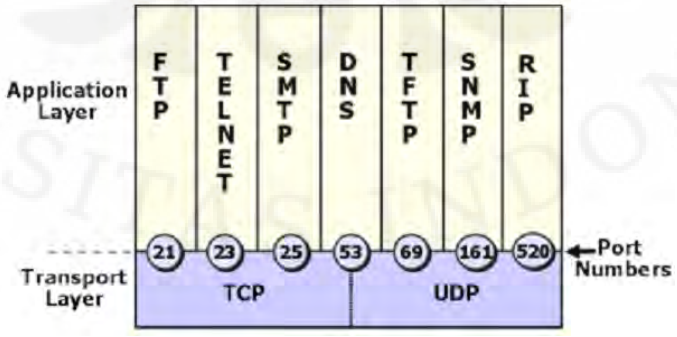


Gambar 2.14 Segment pada UDP [8]

Protokol-protokol yang menggunakan UDP diantarnya adalah TFTP, SNMP, Network File System (NFS), dan Domain Name system (DNS).

2.2.3 Port Numbers

Nomor port (*Port Number*) adalah suatu nomor identifikasi yang digunakan oleh TCP dan UDP untuk menghantarkan informasi menuju ke layer atas (*upper layers*). Nomor port berfungsi untuk melacak pembicaraan yang berbeda didalam network dalam waktu yang bersamaan. Para developer aplikasi software sepakat untuk menggunakan "*well-known port numbers*" yang dikontrol oleh Internet Assigned Numbers Authority (IANA). Sebagai contoh, setiap pembicaraan melalui aplikasi FTP menggunakan standard port number 21. Pembicaraan yang tidak melibatkan suatu aplikasi dengan suatu *well-known port number* adalah port number yang ditugaskan secara acak, dipilih dari jarak tertentu. Port number ini digunakan sebagai sumber (source) dan alamat tujuan ada segment TCP.



Gambar 2.15 Port Number untuk beberapa protokol [8]

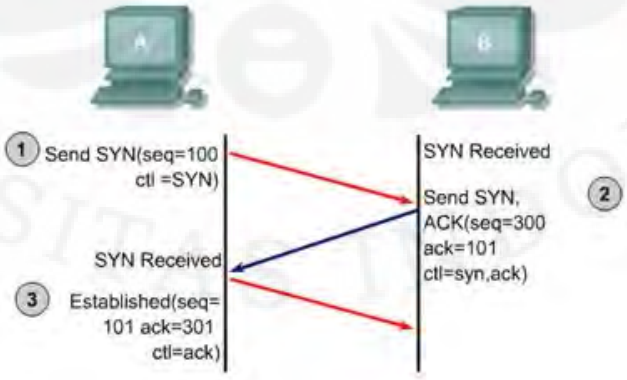
Ada beberapa aplikasi yang tidak dapat mendukung port-port yang telah disediakan oleh TCP dan UDP, port number tersebut memiliki range diantara:

a. Nomor dibawah 1024 adalah "*well-known port*"
b. Nomor diatas 1024 adalah "*dynamically assigned port*"
c. Registered port, adalah port yang terdaftar oleh vendor aplikasi tertentu dan pada umumnya diatas 1024.

DNS menggunakan UDP sebagai solusi untuk menemukan nama pengenal, dan TCP sebagai zona transfer server.

2.2.4 Koneksi TCP

TCP termasuk dalam kategori *connection-oriented*, sehingga memerlukan terbentuknya koneksi terlebih dahulu sebelum transfer data dilakukan. Dua host yang akan saling berkomunikasi akan melakukan proses *Initiating Sequence Number*. Selama transmisi data, dua host yang berkomunikasi terlebih dahulu melakukan proses sinkronisasi membentuk koneksi virtual untuk setiap sesi antar host. Proses sinkronisasi ini memastikan bahwa kedua sisi siap untuk transmisi data dan memperbolehkan host untuk menentukan nomor urutan awal untuk sesi tersebut. Proses ini di kenal sebagai "*Three-way handshake*". Ini merupakan proses *three-step* yang membentuk koneksi virtual antar dua buah host. Penting juga untuk dicatat bahwa three-way handshake dimulai oleh host client. Untuk membentuk sebuah sesi TCP, host client akan menggunakan nomor port layanan yang dikenal untuk dihubungi pada sebuah host server.



Gambar 2.16 Proses "Three-way Handshake" pada TCP [8]

Pada langkah pertama, *initiating host* (client) mengirim sebuah paket sinkronisasi (SYN flag set) untuk melakukan inisialisasi koneksi. Ini menandakan bahwa sebuah paket memiliki nilai awal Sequence Number yang valid dalam segmen ini untuk sesi x. Bit SYN pada bagian header menandakan sebuah permintaan koneksi. Bit SYN merupakan bit tunggal pada field code dari header TCP segment. Sequence Number adalah 32 bit field header TCP segment.

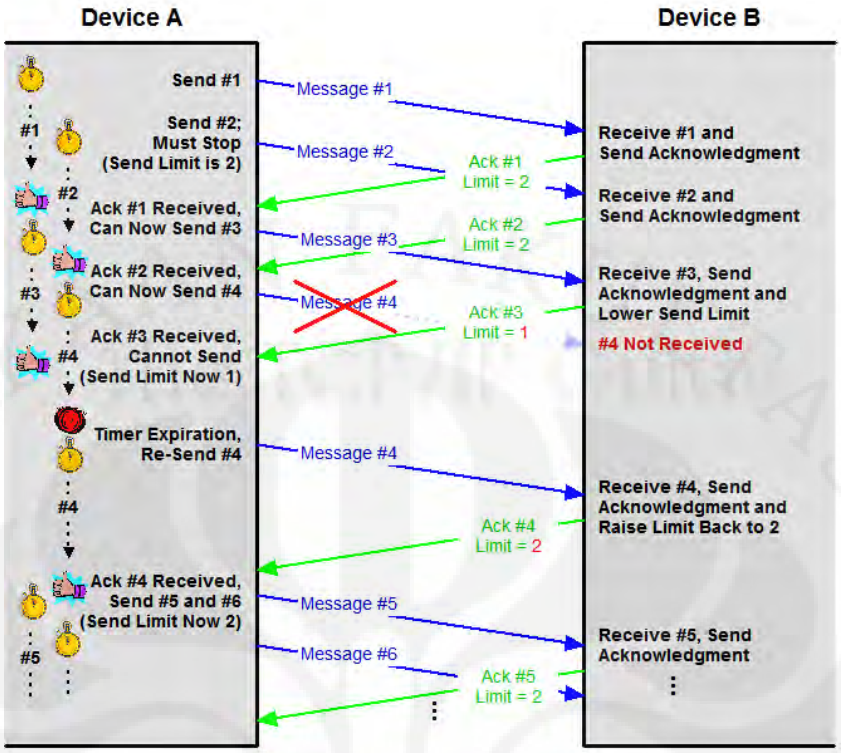
Pada langkah kedua, host lain menerima paket, mencatat Sequence Number x dari client, dan membalas dengan sebuah acknowledgment (ACK flag set). Kumpulan bit control ACK menandakan field Acknowledgment Number berisi sebuah nilai acknowledgment valid. ACK flag merupakan bit tunggal pada kode field dari header TCP segment dan Acknowledgment Number adalah 32 bit field header TCP segment. Sekali koneksi terbentuk, flag ACK di set untuk semua segment selama sesi. Field acknowledgment number berisi sequence number berikutnya yang diharapkan diterima host ini (x+1). Acknowledgment number x+1 berarti host telah menerima semua byte termasuk x, dan mengharapkan menerima byte x+1 berikutnya. Host juga mengajukan sebuah return session. Ini termasuk sebuah segment TCP dengan nilai awal sequence numbernya sendiri dari y dan dengan sekumpulan flag SYN.

Pada langkah ketiga, initiating host merespon dengan sebuah nilai Acknowledgment number dari y+1, yang mana nilai dari sequence number host B+1. Ini menandakan bahwa telah diterima acknowledgment sebelumnya dan mengakhiri proses koneksi untuk sesi ini. Sangat penting untuk memahami bahwa *initial sequence number* digunakan untuk mengajukan komunikasi antar dua peralatan, nomor ini bertindak sebagai nilai referensi awal antar dua peralatan. Sequence number memberikan setiap host sebuah cara untuk menyatakan sehingga penerima mengetahui pengirim merespon untuk permintaan koneksi yang benar.

2.2.5 TCP Window Size

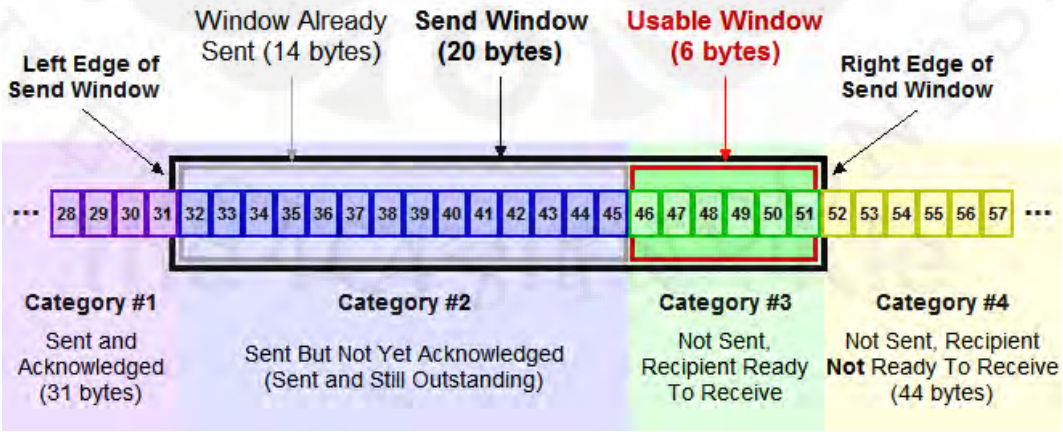
Jumlah data yang harus dikirimkan melalui jaringan sering kali terlalu besar untuk dikirimkan dengan sebuah segmen tunggal, oleh karena itu data perlu dipecah-pecah menjadi beberapa segmen. Dalam TCP ada pengaturan yang menentukan bagaimana segmen-segmen ini dikirim ke penerima. Tidak ada jaminan bahwa spesifikasi perangkat yang berperan sebagai pengirim akan sama dengan perangkat penerima, kemampuan masing-masing perangkat ini untuk mengirim dan menerima data bisa jadi berbeda. Oleh karena itu TCP memiliki sebuah mekanisme untuk mengatur peredaran data dari host sumber ke host tujuan, mekanisme ini disebut dengan *flow control*. Mekanisme memastikan bahwa pengirim mengirim data dengan jumlah dan kecepatan yang memungkinkan untuk diterima disisi penerima. Proses pengendalian aliran data ini disebut dengan *Windowing.*

Seperti yang telah dijelaskan pada bagian sebelumnya bahwa data yang dikirim pada TCP akan dibalas dengan sebuah *acknowledgement* yang memastikan bahwa data sudah diterima. Permasalahannya adalah, terkadang acknowledgement itu sendiri juga bisa hilang ketika ditransmisikan didalam jaringan. Sehingga misalkan suatu perangkat A mengirim data ke perangkat B yang sudah diterima perangkat B, dan lalu B mengirimkan acknowledgement ke perangkat A. Jika acknowledgement ini hilang maka perangkat A akan terus menerus menunggu B untuk dapat mengirim paket selanjutnya. Hal ini sangat membuang-buang waktu, oleh karena itu diterapkan sebuah aturan dimana perangkat A memiliki ketetapan batas waktu yang digunakan untuk menunggu balasan acknowledgement dari perangkat B. waktu interval ini biasanya dihitung dengan memperkirakan lamanya waktu pengiriman data, dan delay alami pada jaringan. Perangkat A hanya akan menunggu acknowledgement dari B selama batas waktu yang diizinkan, jika sampai melebihi batas waktu tersebut masih belum ada balasan, maka perangkat A akan berasumsi bahwa paket yang dikirim tidak sampai ke penerima sehingga perangkat A akan mengirimkannya sekali lagi. Hal ini disebut dengan *Positive Acknowledgement with Retransmision (PAR)*, yang diilustrasikan pada halaman berikutnya.



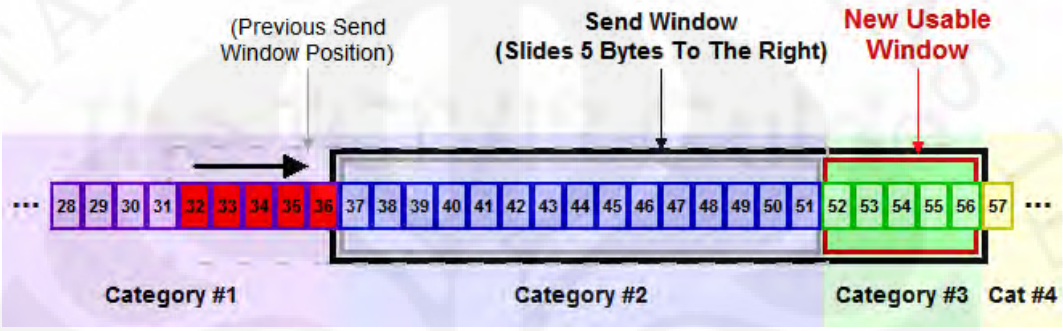
Gambar 2.17 Mekanisme *Positive Acknowledgement with Retransmision (PAR)*[9]

TCP Sliding Window mengambil ide yang sama dengan mekanisme diatas, namun dengan beberapa penyesuaian. Jika pada PAR pesan yang dikirim diberi nomor ID, sementara pada TCP paket yang dikirim diberi *sequence number*. *Acknowledgement* diberikan pada setiap satu *sequence*, bukan byte. Jumlah byte yang diizinkan penerima untuk ditransmisikan dari pengirim sebelum menerima acknowledgement inilah yang disebut *Send Window* atau sering disingkat menjadi *Window* saja.



Gambar 2.18 Struktur TCP Window [9]

Window menentukan berapa banyak byte yang dapat ditransmisikan oleh perangkat pengirim, besarnya sama dengan jumlah kategori 2 (paket terkirim namun belum ada acknowledgement) dan kategori 3 (belum dikirim tapi siap untuk ditransmisikan). Ketika pengirim menerima acknowledgement, maka perangkat ini sudah dapat memindahkan byte di kategori 3 kekategori 2. Terjadi pergeseran karena byte dari kategori 2 pindah ke kategori 1, dan dari kategori 4 ke kategori 3. Pergeseran ini pada akhirnya disebut sebagai *Slides*.



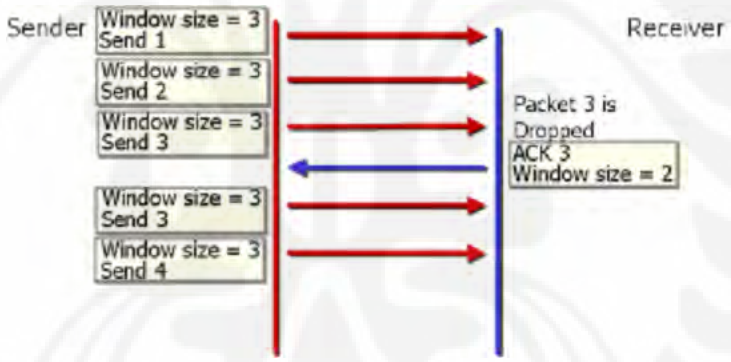
Gambar 2.19 Pergeseran (*Slides*) pada TCP Window [9]

Ukuran window (*window size*) menentukan seberapa banyak data yang akan ditransmisikan pada suatu waktu. Dengan satu ukuran window, setiap segment harus saling balas sebelum segment lain ditrasmisikan, yang akan menghasilkan efisiensi penggunaan bandwidth oleh setiap host. Ukuran window TCP merupakan variable selama masa koneksi. Setiap acknowledgement terdiri dari sebuah window advertisement menandakan seberapa banyak byte yang dapat diterima oleh si penerima.

Paket yang diterima tidak selalu bisa langsung diteruskan ke tingkat aplikasi begitu saja, data yang diterima disimpan terlebih dahulu didalam *buffer*. Ada kalanya penerima harus mengatur aliran paket yang diterimanya, misalkan penerima mendapat paket dengan window size 140, namun hanya mampu meneruskan ke tingkat aplikasi sebesar 40, maka sisa 100 byte masih harus disimpan didalam buffer. Alasan lain adalah terkadang komunikasi yang berlangsung tidak hanya terjadi antara dua perangkat saja, misalkan transmisi dilakukan dari client ke server, bisa jadi pada saat yang bersamaan server juga melayani komunikasi dengan perangkat lain. Akibatnya buffer server bisa masih terisi data saat tiba waktunya paket lain untuk datang, jika kondisi ini berlanjut

maka efeknya adalah data tidak jadi diterima. TCP memiliki kemampuan untuk menangani kontrol kemacetan (*congestion control*) window, yang memiliki ukuran sama dengan window penerima, tetapi akan terpotong setengahnya jika sebuah segment hilang, contohnya ketika terjadi kemacetan (*congestion*). Pendekatan ini memperbolehkan window untuk memperbesar ukurannya apabila perlu untuk mengatur ruang buffer dan pemrosesan. Semakin besar ukuran window maka semakin banyak data yang dapat diproses.

Sebagai contoh, sebuah host pengirim mencoba untuk mengirim data dengan ukuran window 3. Namun ketika data sampai ditujuan, host penerima ternyata hanya bisa menangani dua dari tiga data yang dikirim oleh karena itu penerima hanya akan mengirimkan kembali dua acknowledgement. Ketika acknowledgement sampai kembali ke pengirim, maka pengirim akan tahu bahwa ia harus mengurangi ukuran windownya menjadi dua agar dapat diterima sepenuhnya oleh host penerima.



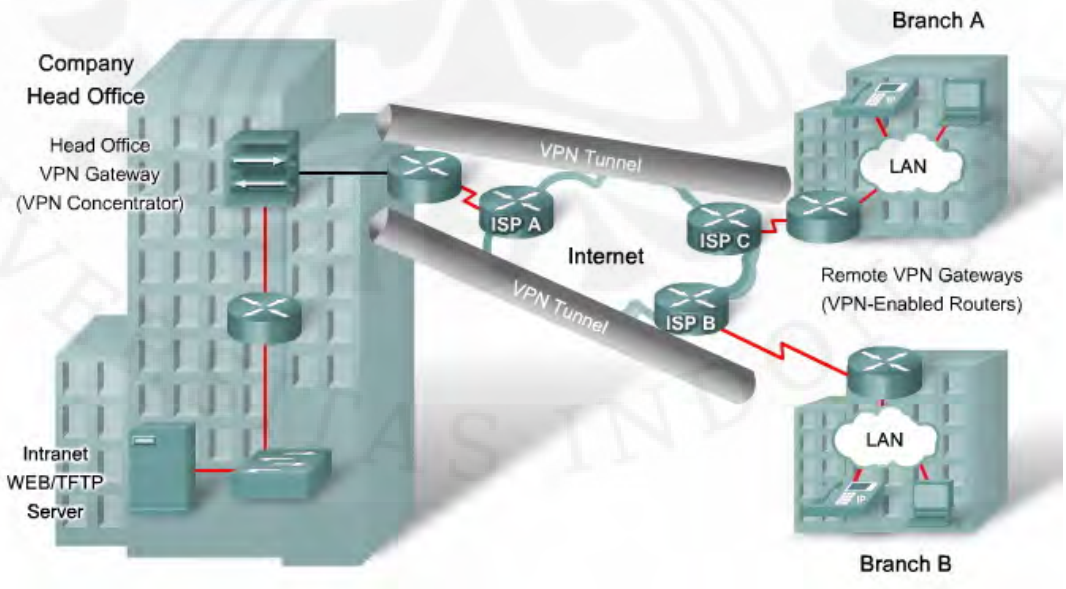
Gambar 2.20 Proses "*Windowing*" pada TCP [8]

2.3 Virtual Private Network

Isu yang kerap kali didengungkan terkait jaringan dengan skala besar seperti WAN adalah permasalahan perfoma dan keamanan. Sering kali ketika kita harus membuat jalur komunikasi antar tempat-tempat yang terpisah cukup jauh maka kita harus menggunakan fasilitas jaringan publik untuk menekan biaya. Jelas penggunaan jaringan publik akan jauh lebih murah ketimbang membangun jaringan sendiri dalam skala geografi yang cukup luas, namun cara ini justru menciptakan kerentanan keamanan dalam komunikasi jaringan. *Virtual Private Network* (VPN) muncul sebagai salah satu solusi untuk menciptakan jaringan

komunikasi yang aman antara dua titik yang terhubung melalui jaringan publik. Ide dasar dari VPN adalah *Tunneling*, sebuah teknologi yang memungkinkan terciptanya saluran privat virtual didalam jaringan publik. Tunnel di dalam dunia jaringan diartikan sebagi suatu cara untuk meng-enkapsulasi atau membungkus paket IP didalam paket IP yang lain. Dimana titik dibelakang IP Tunnel akan memberikan paket IP melalui Tunnel yang dibuat dan mengirimkannya ke sebuah titik dibelakang tunnel yang lain. Intinya tunneling adalah suatu cara membuat jalur private dengan menggunakan infrastruktur pihak ketiga. Ketika sebuah paket IP dapat dicapai oleh masing-masing sisi client dibelakang IP tunnel, maka Tunnel IP Header dan beberapa Tunel Header tambahan yang membungkus paket IP tersebut akan dilepas dan Paket IP yang asli akan disuntikan ke dalam IP Stack pada titik dibelakang IP Tunnel. Paket yang dikirim didalam tunnel ini di enkripsi dengan suatu format tertentu oleh protokol VPN lalu dikirim dari client ke server pada VPN tunnel. Ketika paket sampai di node ujung tunnel, kemudian paket ini akan di dekripsi terlebih dahulu sebelum diteruskan ke layer atas protokol TCP/IP.

Proses enkripsi-dekripsi pada VPN yang membutuhkan waktu akan menambah delay pada jaringan, namun bagaimanapun VPN memberikan kemanan dalam komunikasi didalam jaringan WAN.

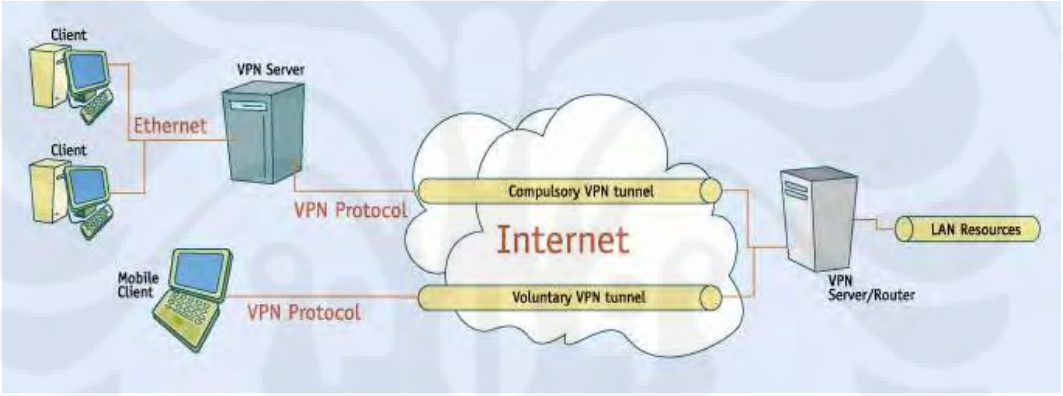


Gambar 2.21 VPN Tunnel pada WAN [5]

2.3.1 Tipe-tipe Tunneling Pada VPN

VPN mendukung dua tipe *tunneling* yang umum digunakan, yaitu *Compulsory* dan *Voluntary Tunneling*. Pada voluntary tunneling, VPN client yang bertindak sebagai pengatur koneksi, remote access client dapat membentuk koneksi langsung dengan VPN server kapanpun client terhubung dengan internet.

Compulsory tunneling dibangun antara dua VPN server yang bertindak sebagai router untuk semua trafiik jaringan. User client yang menggunakan tunel ini bisa jadi tidak menyadari bahwa koneksi yang sedang digunakannya adalah koneksi aman yang terenkripsi, tipe ini sangat cocok untuk menghubungkan dua LAN ditempat terpisah, seperti menghubungkan kantor-kantor cabang sebuah perusahaan.



Gambar 2.22 VPN dengan dua tipe tunneling [10]

2.3.2 Jenis-jenis Protokol VPN

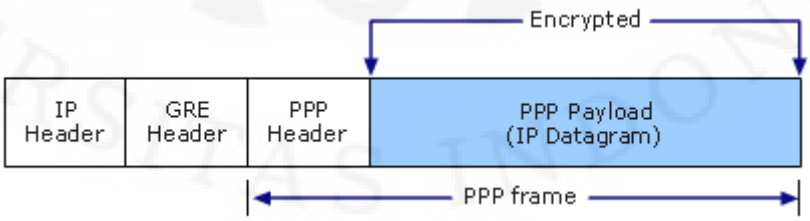
Ada beberapa jenis protokol yang dapat diimplementasikan pada VPN tunel, yaitu :

1. **Point-to-Point Tunneling Protocol (PPTP)**

   Beberapa perusahan bekerja sama mengembangkan spesifikasi protokol jaringan yang digunakan dalam implementasi VPN ini. Point-to-Point Tunneling Protocol (PPTP) adalah suatu protokol jaringan yang memungkinkan pengiriman data secara aman dari remote client kepada server perusahaan swasta dengan membuat suatu virtual private network (VPN)

melalui jaringan data berbasis TCP/IP. Teknologi jaringan PPTP merupakan perluasan dari remote access Point-to-Point protocol yang telah dijelaskan dalam RFC 1171 yang berjudul "*The Point-to-Point Protocol for the Transmission of Multi-Protocol Datagrams over Point-to-Point Links*" . PPTP adalah suatu protokol jaringan yang membungkus paket PPP ke dalam IP datagram untuk transmisi yang dilakukan melalui internet atau jaringan publik berbasis TCP/IP. PPTP dapat juga digunakan pada jaringan LAN-to-LAN. Protokol PPTP termasuk dalam sistem operasi server Windows NT versi 4.0 dan Windows NT Workstastion versi 4.0. Komputer yang menjalankan sistem operasi ini dapat dengan menggunakan protokol PPTP untuk koneksi ke jaringan private sebagai remote access client secara aman melalui jaringan publik seperti internet. Dengan kata lain, PPTP digunakan sesuai dengan permintaan, misalnya dapat digunakan dengan VPN melalui internet atau jaringan publik berbasis TCP/IP lainnya. PPTP juga dapat digunakan pada LAN untuk membuat VPN dalam LAN. Fitur penting dalam penggunaan PPTP adalah, PPTP mendukung VPN dengan menggunakan Public-Switched Telephone Networks (PSTNs). PPTP menyederhanakan dan mengurangi biaya dalam penggunaan pada perusahaan besar dan sebagai solusi untuk remote atau mobile users karena PPTP memberikan komunikasi yang aman dan terenkripsi melalui line public telephone dan internet.

PPTP mengenkapsulasi frame PPP dalam sebuah IP datagram. PPTP menggunakan koneksi TCP untuk pengaturan tunel dan memodifikasi *Generic Routing Encapsulation* (GRE) untuk mengenkapsulasi frame PPP untuk data yang dikirim melalui tunel.

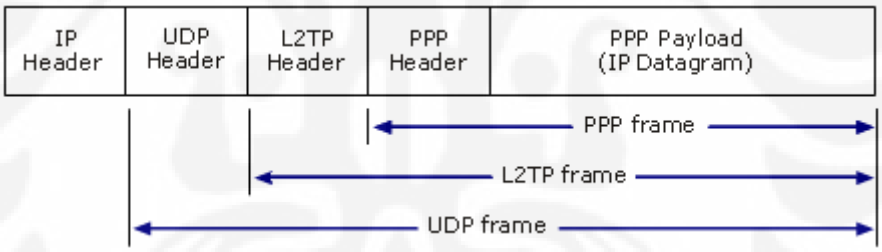


Gambar 2.23 Enkapsulasi pada protokol PPTP [11]

Paket frame PPP dienkripsi dengan Microsoft Point-to-Point Encryption (MPPE) menggunakan kunci enkripsi yang dihasilkan dari MS-CHAP v2 atau proses autentikasi EAP-TLS. Client VPN harus memilih diantara MS-CHAP v2 atau protokol autentikasi EAP-TLS untuk dapat mengirimkan data dalam bentuk terenkripsi.

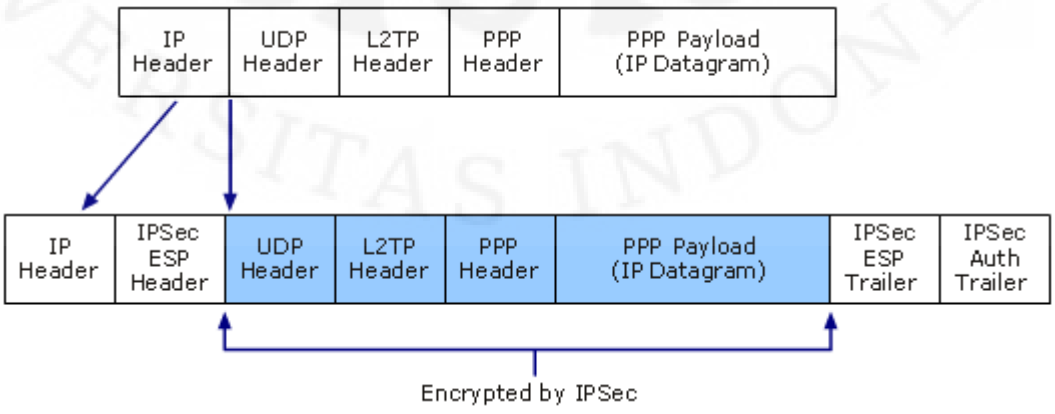
2. Layer Two Tunneling Protocol (L2TP)

Kompetitor utama dari PPTP sebenarnya adala L2F, sebuah protokol yang diimplementasikan pada produk-produk Cisco. Usaha pengembangan L2F lalu berujung pada pengkombinasian keunggulan L2F dengan keunggulan PPTP, menjadi sebuah protokol baru yaitu L2TP. Protokol L2TP sering juga disebut sebagai protokol dial-up virtual, karena L2TP memperluas suatu session PPP (Point-to-Point Protocol) dial-up melalui jaringan publik internet, atau sering juga digambarkan seperti koneksi virtual PPP. Struktur paket L2TP ditunjukkan seperti gambar berikut:



Gambar 2.24 Struktur Paket L2TP berisi IP Datagram [11]

Proses enkapsulasi pada L2TP sama dengan IPSec, terlihat pada diagram berikut ini:



Gambar 2.25 Enkripsi pada L2TP/IPSec [11]

Enkapsulasi pada L2TP/IPSec terdiri dari dua layer yaitu:

a. Layer pertama : Enkripsi L2TP, paket frame PPP dibungkus dengan sebuah L2TP Header, dan kemudian ditambah dengan UDP Header. Gambar dihalaman sebelumnya menunjukkan hal ini.
b. Layer kedua : Enkapsulasi IPScec, hasil akhir dari enkapsulasi L2TP adalah pesan yang telah dibungkus dengan sebuah header tambahan yaitu IPSec Encapsulating Security Payload (ESP) header ditambah dengan tailer pada akhir frame. IPSec Authentication Tailer menyediakan autentikasi dan integritas data dan IP header final. Pada IP header disimpan informasi mengenai sumber dan tujuan yang berfungsi sebagai client dan server.

Data L2TP dapat dienkripsi dengan Data Encryption Standard (DES) ataupun Triple DES (3DES) menggunakan kunci yangg dihasilkan dari Internet key Exchange (IKE).

3. Internet Protocol Security (IPSec)

IPSec adalah sebuah protokol yang digunakan untuk mengamankan transmisi datagram dalam sebuah internetwork berbasis TCP/IP. IPSec mendefiniskan beberapa standar untuk melakukan enkripsi data dan juga integritas data pada lapisan kedua dalam *DARPA Reference Model* (internetwork layer). IPSec melakukan enkripsi terhadap data pada lapisan yang sama dengan protokol IP dan menggunakan teknik tunneling untuk mengirimkan informasi melalui jaringan Internet atau dalam jaringan Intranet secara aman. IPSec didefinisikan oleh badan Internet Engineering Task Force (IETF) dan diimplementasikan di dalam banyak sistem operasi. Windows 2000 adalah sistem operasi pertama dari Microsoft yang mendukung IPSec. IPSec diimplementasikan pada lapisan transport dalam OSI Reference Model untuk melindungi protokol IP dan protokol-protokol yang lebih tinggi dengan menggunakan beberapa kebijakan keamanan yang dapat dikonfigurasikan

untuk memenuhi kebutuhan keamanan pengguna, atau jaringan. IPSec umumnya diletakkan sebagai sebuah lapsian tambahan di dalam *stack* protokol TCP/IP dan diatur oleh setiap kebijakan keamanan yang diinstalasikan dalam setiap mesin komputer dan dengan sebuah skema enkripsi yang dapat dinegosiasikan antara pengirim dan penerima. Kebijakan-kebijakan keamanan tersebut berisi kumpulan filter yang diasosiasikan dengan kelakuan tertentu. Ketika sebuah alamat IP, nomor port TCP dan UDP atau protokol dari sebuah paket datagram IP cocok dengan filter tertentu, maka perlakuan yang terkait dengannya akan diaplikasikan terhadap paket IP tersebut. Peraturan-peraturan dalam ketentuan IPSec tersebut digunakan untuk memulai dan mengontrol komunikasi yang aman berdasarkan sifat lalu lintas IP, sumber lalu lintas tersebut dan tujuannya. Peraturan-peraturan tersebut dapat menentukan metode-metode autentikasi dan negosiasi, atribut proses tunneling, dan jenis koneksi.

2.4 Pengukuran Performa Jaringan (QoS)

*Quality of Service (*QoS) merupakan kemampuan jaringan untuk menyediakan service yang lebih baik pada suatu trafik tertentu mulai berbagai macam teknologi meliputi jaringan IP, frame relay, ATM dan SDH. Elemen QoS tergantung dari informasi yang ditransmisikan (voice, data atau video). Faktor yang mempengaruhi QoS pada jaringan IP yang dibatasi pada masalah seperti berikut :

1. ***Availability***, yaitu persentase hidupnya sistem atau subsistem telekomunikasi. Idealnya, availability harus mencapai 100 %. Nilai availability yang diakui cukup baik adalah 99,9999 % (six nines), yang menunjukkan tingkat kerusakan sebesar 2,6 detik per bulan [12].
2. ***Throughput***, yaitu kecepatan (rate) transfer data efektif, yang diukur dalam bps. Header dalam paket data mengurangi nilai ini. Throughput dapat dihitung dengan melihat jumlah paket yang datang terhadap yang dikirim. Throughput (S) adalah total waktu yang

digunakan untuk mengirim paket dengan sukses per satuan waktu tertentu yang dapat dihitung dengan:

$$S = \frac{Jumlah\ paket\ sukses\ x\ waktu\ transmisi\ paket}{Lama\ pengalamatan}$$

Offered Traffic (G) adalah total waktu paket yang ditawarkan persatuan waktu yang dapat dihitung dengan :

$$G = \frac{Jumlah\ paket\ muncul\ x\ waktu\ transmisi\ paket}{Lama\ pengalamatan}$$

Sedangkan waktu transmisi paket (t trans) terdiri dari dua komponen yaitu waktu paket (t paket) dan delay propagasi (t prop).

t transmisi = t paket + t prop

3. ***Jitter***, merupakan masalah khas dari *connectionless network* atau *packet switched network*. Jitter didefinisikan sebagai variasi delay yang diakibatkan oleh panjang queue dalam suatu pengolahan data dan reassemble paket-paket data di akhir pengiriman akibat kegagalan sebelumnya. Secara umum jitter merupakan masalah dalam slow speed links.

Tabel 2.2 Klasifikasi Kualitas Jitter [12]

Kategori Degradasi	**Peak Jitter**
Sangat Bagus	0 ms
Bagus	75 ms
Sedang	125 ms
Jelek	225 ms

Diharapkan bahwa peningkatan QoS dengan mekanisme priority buffer, bandwidth reservation (RSVP, MPLS dll) dan high speed connections dapat mereduksi masalah jitter di masa yang akan datang. Jitter diantara titik awal dan akhir komunikasi seharusnya kurang dari 150 ms sedangkan untuk wireless kurang dari 5 ms [12]

4. ***Packet*** Loss, komunikasi real time didasari oleh protokol UDP dimana bersifat connectionless dan jika paket gagal dikirim maka paket tersebut tidak akan dikirim lagi dan menjadi masalah besar jika packet loss yang terjadi sangat besar. Packet loss untuk aplikasi voice dan multimedia dapat ditoleransi sampai dengan 20% untuk single Access Point [12]. Packet Loss dapat dihitung dengan:

$$\text{Packet Loss} = \frac{(Jumlah\ data\ dikirim - Jumlah\ data\ diterima)}{Jumlah\ data\ dikirim} \times 100$$

Tabel 2.3 Klasifikasi Kualitas Packet Loss [12]

Kategori Degradasi	Packet Loss
Sangat Bagus	0 %
Bagus	3 %
Sedang	15 %
Buruk	25 %

5. ***Delay***, adalah waktu tunda saat paket yang diakibatkan oleh proses transmisi dari satu titik lain yang menjadi tujuannya. Packet delay dapat menyebabkan kualitas suara menjadi turun. Jika delay tidak diminimalkan maka sinyal suara yang diterima akan menyebabkan kualitas yang buruk akibat dari akumulasi seluruh delay yang terjadi di dalam jaringan. Ada dua penyebab terjadinya delay yang diklasifikasikan sebagai *Fixed Delay*, dan *Variable Delay*.

Tabel 2.4 Komponen Delay [12]

Jenis Delay	Keterangan
Alghoritmic Delay	Delay ini disebabkan oleh standar *codec* yang digunakan.
Packetization Delay	Delay yang diakibatkan oleh pengakumulasian bit *voice sample* ke *frame*.
Serialization Delay	Delay ini terjadi karena adanya waktu yang dibutuhkan untuk pentransmisian paket IP dari sisi *originating* (pengirim).
Propagation Delay	Delay ini terjadi karena perambatan atau perjalanan paket IP dimedia transmisi ke alamat tujuan. Contohnya delay propagasi didalam kabel akan memakan waktu 4 sampai 6 µs per kilometernya.
Component Delay	Delay ini disebabkan oleh karena banyaknya komponen yang digunakan didalam sistem transmisi.

Rekomendasi ITU G.114 merekomendasikan standar delay, dimana ada tiga klasifikasi berdasarkan cakupan besaran delay seperti ditunjukkan pada tabel berikut ini:

Tabel 2.5 Rekomendasi ITU-T G.114 untuk delay [12]

Range in Milisecon	Description
0 – 150 ms	Acceptable for most user application.
150 – 400 ms	Acceptable provided that administrators are aware of the transmision time and its impact on transmision quality of user application.
> 400 ms	Unacceptable for general network planning purpose, it is recoqnized that in some exceptional cases thi limit will be exeeded

2.5 Software Simulator OPNET

Ada beberapa *Network Simulator Software* yang beredar saat ini, diantaranya adalah NS-2, Packet Tracer dari Cisco System, OPNET, dll. Beberapa diantara network simulator itu seperti NS-2 bekerja berdasarkan urutan kejadian satu persatu (*discrete event*) kelemahan cara kerja ini adalah waktu yang dibuthkan untuk melakukan simulasi sangat lama, sehingga tentu saja tidak efisien. Simulator lainnya seperti Packet Tracer dari Cisco juga dapat digunakan untuk melakukan simulasi jaringan, namun fitur-fitur dalam software ini sangat terbatas, hanya mampu mensimulasikan beberapa jenis routing protokol, dan traffik data didalam jaringan. OPNET menjadi pilihan karena OPNET merupakan sebuah software simulator berlisensi yang sangat handal untuk melakukan simulasi dan analisis performa jaringan komputer. Software ini memungkinkan kita untuk membuat simulasi jaringan komputer yang melibatkan berbagai unsur teknologi jaringan.

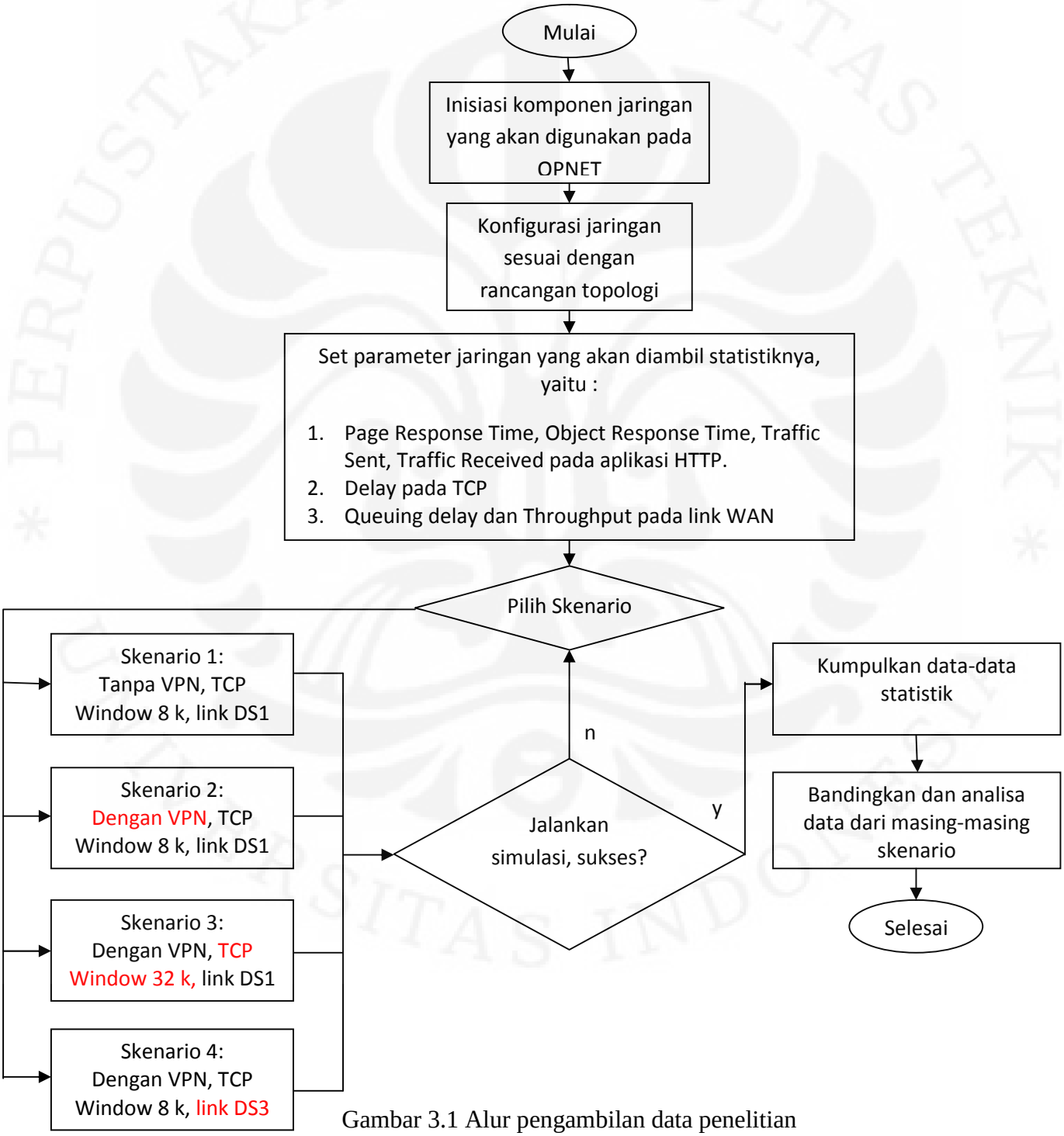
Gambar 2.26 Software Simulator OPNET IT Guru Academic Edition 9.1

OPNET menyediakan begitu banyak fitur-fitur teknologi jaringan terkini, mulai dari perangkat dari berbagai macam vendor, protokol, applikasi, dan semua hal yang berhubungan dengan jaringan itu sendiri. Software ini sangat berguna untuk merancang jaringan, menguji performa jaringan maupun aplikasi, pengujian hardware, dan lain-lain. OPNET menyediakan suatu lingkungan virtual yang lengkap bahkan dengan karakteristik yang sama persis dengan jaringan real, OPNET dapat menghasilkan berbagai macam output yang ditampilkan secara praktis melalui visualisasi statistik berupa grafik-grafik, dan bahkan juga ada *Log* yang merekam aktifitas jaringan sehingga kita dapat mengetahui jejak perjalanan trafik, kondisi *error* apa saja yang terjadi, dan semua mengenai operasi jaringan. OPNET menyediakan sebuah fitur canggih untuk menganalisa performa aplikasi yang berjalan pada jaringan, sehingga bisa dilakukan pengujian lebih lanjut untuk memahami interaksi antara aplikasi, server, dan jaringan. Fitur ini disebut *Application Characterization Environtment (ACE)*, yang memungkinkan kita untuk menangkap, mem-filter, dan mensinkronisasi jejak-jejak aplikasi dari berbagai segmen jaringan. Sehingga penelitian mengenai network, perangkat-perangkat didalamnya, dan protokol, serta teknologi lain yang berhubungan dengan jaringan dapat dilakukan secara komprehensif dan mendalam tanpa perlu lagi memerlukan biaya yang mahal untuk membeli peralatan jaringan, dan tidak diperlukan lagi ruangan laboratorium yang besar untuk melakukan penelitian dibidang ini, semua cukup dilakukan dalam sebuah software yang handal yang dapat melakukan simulasi apapun dibidang networking. OPNET tersedia dalam versi gratis khusus untuk keperluan penelitian akademis di universitas, versi gratis ini dinamakan OPET IT Guru Academic Edition yang dapat diunduh dihalaman berikut ini (http://www.opnet.com/university_program/itguru_academic_edition/).

Meskipun OPNET IT Guru hadir dengan beberapa keterbatasan dibandingkan dengan OPNET Modeler, namun IT Guru sudah cukup bagus dan lengkap untuk melakukan berbagai simulasi. Sebaiknya software simulator ini di implementasikan penggunaannya dilab-lab universitas untuk membantu para mahasiswa memahami konsep networking lebih dalam dan lebih luas, mahasiswa dapat melakukan simulasi dan analisis yang cakupannya lebih luas tanpa perlu mengeluarkan biaya yang besar.

BAB 3
SIMULASI VPN PADA WAN DAN METODE PENGAMBILAN DATA

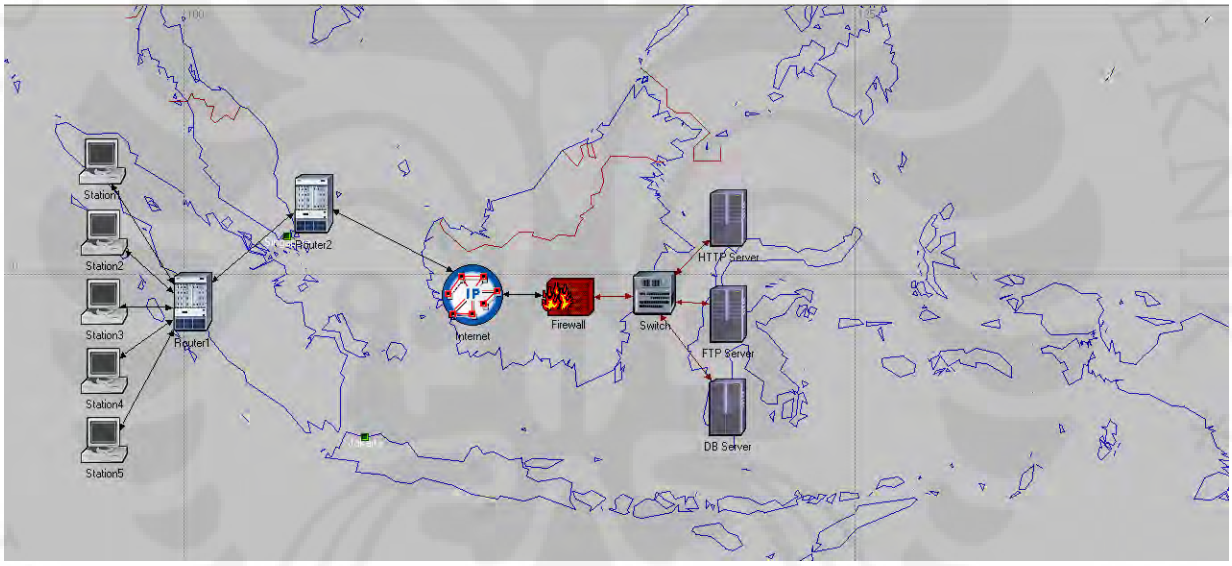
Simulasi ini dilakukan menggunakan software simulator OPNET IT Guru Academic Edition 9.1. Dalam satu *project* terdapat beberapa skenario yang disesuaikan dengan keperluan pengambilan data untuk menguji performa aplikasi pada jaringan VPN. Berikut ini alur pengambilan data penelitian:



Gambar 3.1 Alur pengambilan data penelitian

3.1 Topologi Jaringan

Simulasi ini menunjukkan sebuah jaringan WAN yang tersebar dibeberapa wilayah di Indonesia, skenario pada simulasi ini menggunakan lima *workstation* yang terhubung ke Router1 dan router ini terhubung ke Router2 sebelum masuk ke internet, jaringan ini terhubung melalui WAN ke *Server Farm* yang diletakkan diwilayah pulau Sumatera melalui jaringan internet publik, lalu workstation akan mengakses aplikasi yang terdapat pada server HTTP yang berlokasi di pulau Sulawesi. Penelitian ini mencoba menguji performa aplikasi *Streaming Multimedia* yang berjalan pada aplikasi HTTP. Berikut ini gambar topologi jaringan yang digunakan :



Gambar 3.2 Topologi Jaringan

3.2 Komponen Jaringan

Komponen jaringan ini adalah sebagai berikut:

- *Workstation*, merupakan komputer PC yang digunakan sebagai *client*, yang menjalankan fungsi PC biasa pada infrastruktur jaringan. Workstation ini akan mengakses aplikasi-aplikasi yang terdapat pada server.
- *Router*, perangkat ini berfungsi sebagai penghubung antar jaringan-jaringan yang berbeda. Router bertugas untuk menyampaikan paket dari

sumber ke tujuan yang terpisah pada jaringan yang berbeda, proses penyampaian paket ini dosebut *routing*.

- *Internet*, merupakan jaringan komputer publik yang merepresentasikan jaringan WAN yang lebih besar.
- *Firewall*, adalah perangkat yang berfungsi untuk memfilter paket yang masuk kedalam jaringan tertentu, firewall dapat memilah-milah jenis paket dan tugasnya mengamankan suatu jaringan dari paket-paket yang tidak diinginkan.
- *Switch*, merupakan penghubung beberapa perangkat untuk membentuk jaringan kecil atau *Local Area Network* (LAN).
- *Server*, berfungsi sebagai perangkat penyedia layanan dalam jaringan komputer.
- *WAN Link*, ada dua jenis WAN link yang digunakan yaitu DS1 dan 100BaseT LAN. DS1 atau yang juga bisa disebut T1 merupakan kabel pembawa sinyal berkecepatan tinggi yang biasa digunakan sebagai standar jaringan telekomunikasi di Amerika dan Jepang, link ini memiliki bandwidth sebesar 1.544Mbit/s. Semenstara 100BaseT adalah link standar berupa kabel *Twisted Pair* yang digunakan pada Ethernet untuk menyalurkan trafik dengan kecepatan 100Mbit/s.

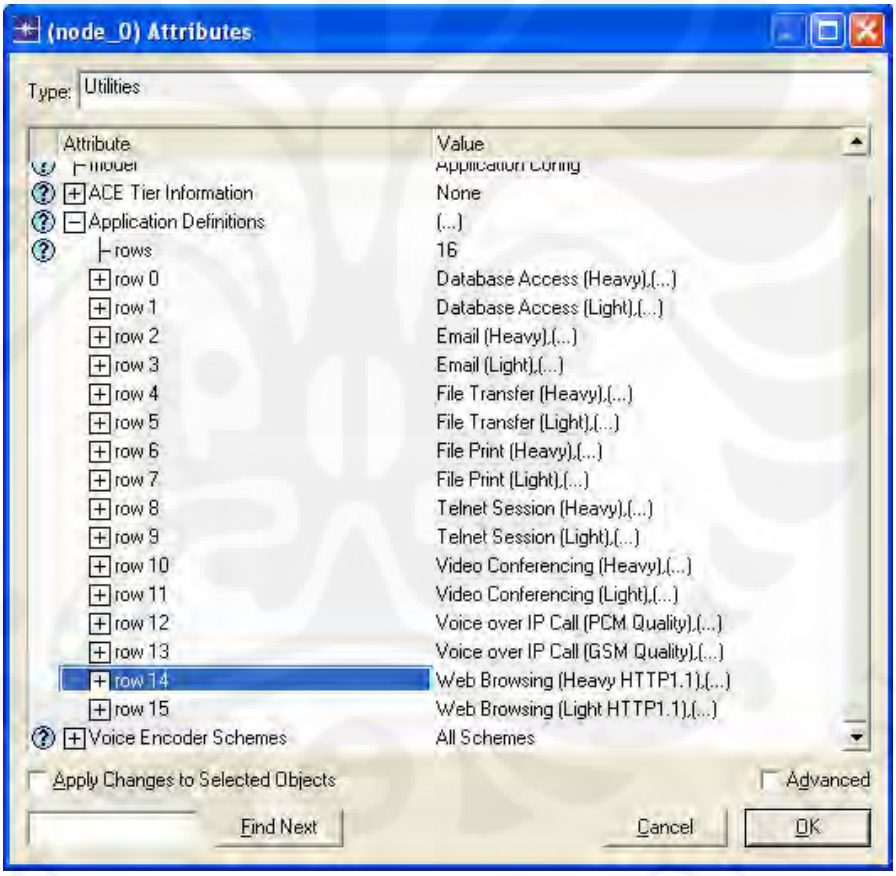
Pada simulator OPNET, komponen diatas didefinisikan pada tabel berikut:

Tabel 3.1 Tabel daftar komponen jaringan pada OPNET

Jumlah	**Nama Komponen**	***Palette***	**Label**
5	ppp_wkstn	internet_toolbox	Station1, Station2, Station3, Station4, Station5
2	ethernet4_slip8_gtwy	internet_toolbox	Router1, Router2
1	ip32_cloud	internet_toolbox	Internet
1	ethernet2_slip8_firewall	internet_toolbox	Firewall
1	ethernet16_switch	internet_toolbox	Switch
3	ethernet_server	internet_toolbox	HTTP Server, FTP Server, DB Server
8	PPP_DS1	internet_toolbox	Tidak diberi label
4	100BaseT	internet_toolbox	Tidak diberi label
1	Application Config	internet_toolbox	Application
1	Profile Config	internet_toolbox	Profile
1	IP Attribute Config	internet_toolbox	IP Attribute

3.4 Konfigurasi Jaringan

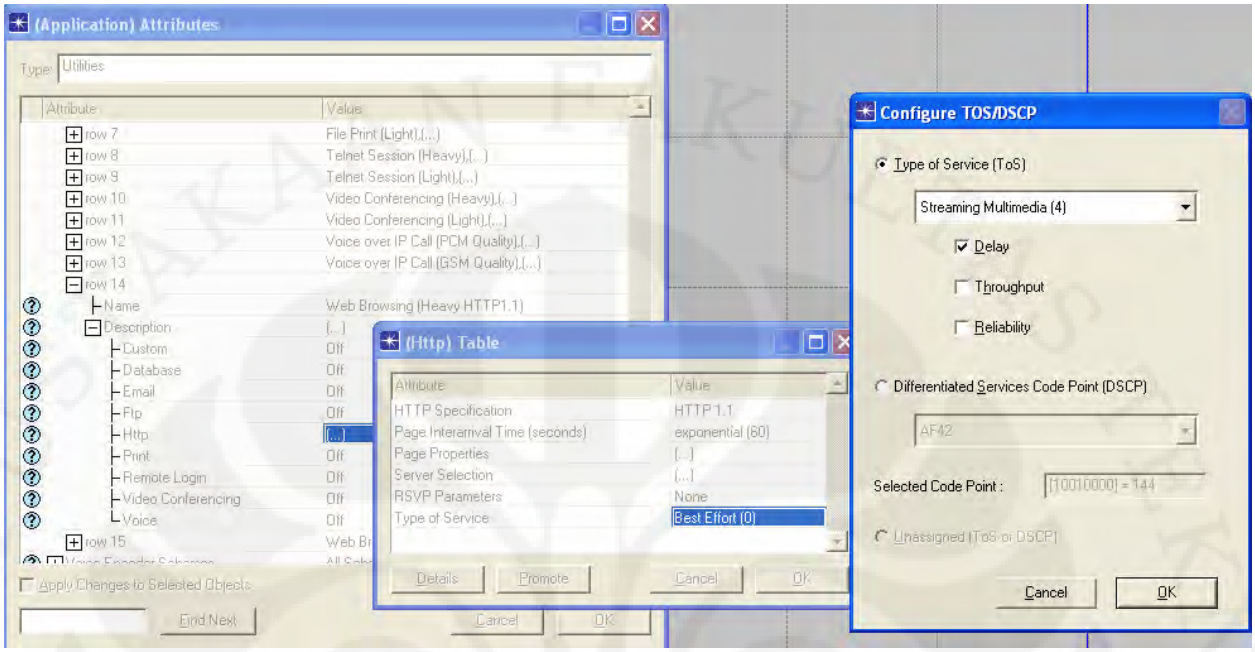
Simulator OPNET memerlukan pendefinisian terhadap parameter-parameter objek yang disebar didalamnya, bersama skripsi ini terlampir panduan dasar bagaimana menggunakan OPNET. Hal yang pertama yang perlu dikonfigurasi adalah aplikasi yang dijalankan pada jaringan ini. Klik kanan pada node **Application** lalu pilih **Edit Attribute**, kemudian akan muncul window baru dan pada **Application Definition** pilih **Default**, hal ini akan menyebabkan OPNET secara otomatis membuat 16 jenis aplikasi. Klik pada tanda plus (+) tepat disamping label Application Definition, akan terlihat ke-enambelas aplikasi yang sudah dibuat dan sorot **row 14** > **Web Browsing (Heavy HTTP1.1)**.



Gambar 3.3 Konfigurasi aplikasi

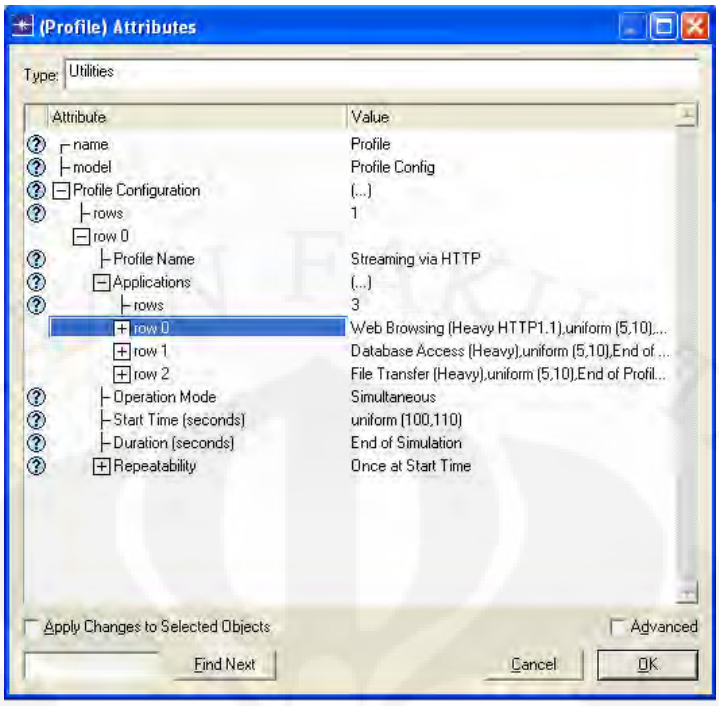
Klik pada tanda plus di **row14** tersebut lalu **Description** > dan pada **Http** pilih **Edit**. Lalu pada baris **Type of Service** yang secara default di-set sebagai **Best Effort (0)** ganti dengan **Streaming Multimedia (4)** selanjutnya tekan **OK**. Hal ini bertujuan untuk mengganti aplikasi yang berjalan pada media HTTP,

dalam penelitian ini kita ingin melihat performa *Streaming Mutlimedia* yang berjalan pada aplikasi HTTP.



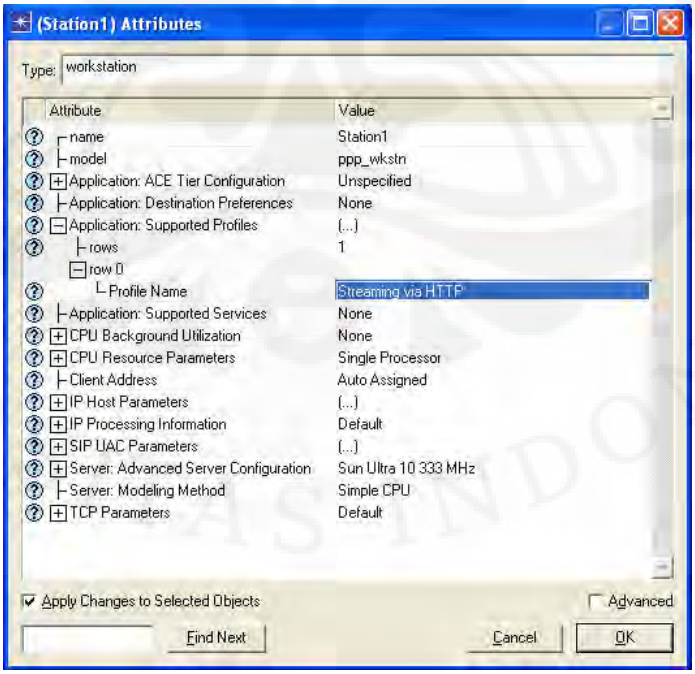
Gambar 3.4 Konfigurasi aplikasi HTTP untuk menjalankan Streaming

Langkah berikutnya adalah mengkonfigurasi node **Profile**, hal ini bertujuan untuk menentukan aplikasi apa saja yang akan dijalankan workstation (Station1 s/d Station5). Klik kanan pada node **Profile**, lalu klik **Edit Attribute** dan pada **Profile Configuration** masukan satu row (pada **rows** pilih **1**). Lalu masukan nama profile yaitu "**Streaming Multimedia via HTTP**" dan pada baris **Application** masukkan tiga jenis aplikasi (pada **rows** pilih **3**), pilih aplikasi **Web Browsing (Heavy HTTP1.1)** pada **row 0**, dan pada dua row berikutnya masukkan **Database Access (Heavy)** dan **File Transfer (Heavy)**. Ubah **Operation Mode** menjadi **Simultaneous** karena aplikasi streaming berjalan secara simultan.



Gambar 3.5 Konfigurasi aplikasi-aplikasi pada Profile

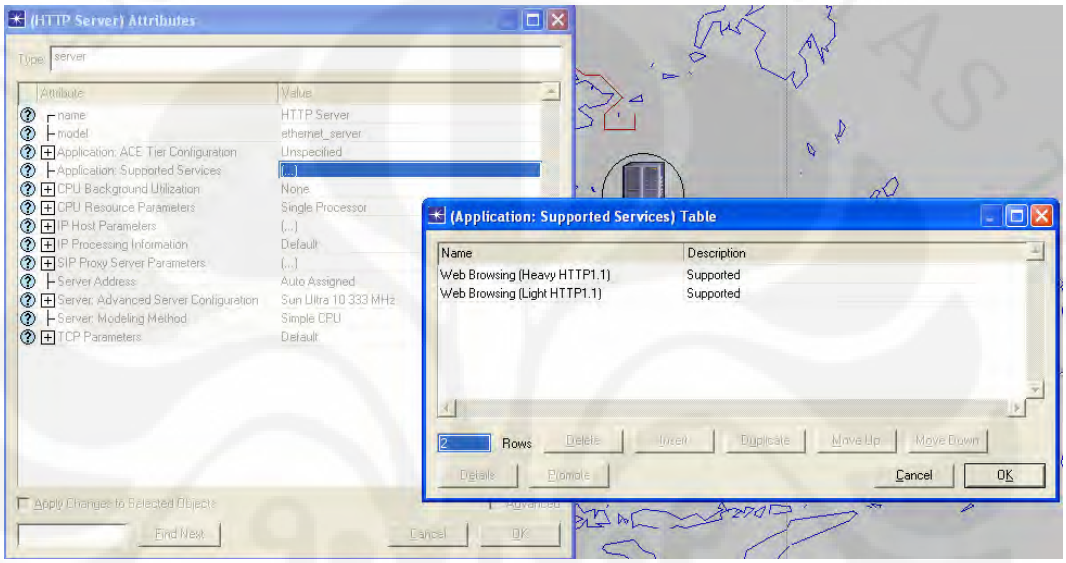
Setelah selesai dengan konfigurasi aplikasi dan profil, selanjutnya adalah konfigurasi workstation yaitu **Station1** sampai **Station5**. Klik kanan pada salah satu dari kelima workstation lalu pilih **Select Similar Nodes** dan pastikan kelima workstation terpilih secara bersamaan. Lalu pada bagian kiri bawah window yang muncul beri tanda pada **Apply Changes to Selected Objects**.



Gambar 3.6 Konfigurasi profil aplikasi yang dijalankan pada workstation

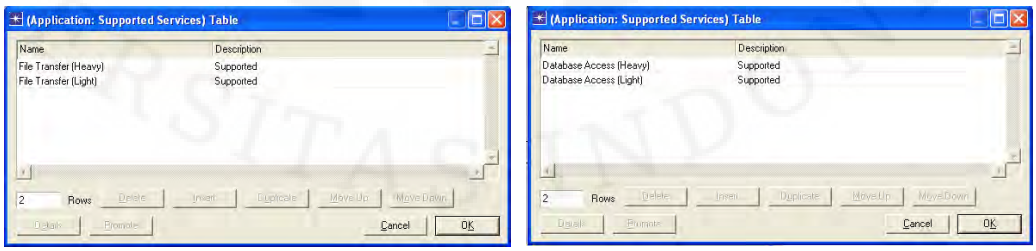
Pada **Application Supported Profiles** isi row dengan angka 1, lalu pada **row 0** pilih **Streaming via HTTP**. Konfigurasi ini akan menyebabkan semua workstation menjalankan aplikasi-aplikasi yang didefinisikan pada profil tersebut.

Berikutnya kita juga perlu mengkonfigurasi Server agar dapat menyediakan layanan aplikasi yang diminta client. Klik kanan pada HTTP Server lalu edit attribute node tersebut sesuai dengan gambar dibawah ini:



Gambar 3.7 Konfigurasi pada HTTP Server

Pada **Application Supported Profiles** pilih **Edit**, lalu pada window yang muncul buat 2 row dan isi dengan **Web Browsing (Heavy HTTP1.1)** dan **Web Browsing (Light HTTP1.1)**. Lakukan hal yang sama pada FTP Server dan DB Server, namun pada FTP Server aplikasi yang dimasukkan adalah **File Transfer (Heavy)** dan **File Transfer (Light)**, sementara pada DB Server masukkan **Database Access (Heavy)** dan **Database Access (Light)**.



Gambar 3.8 Konfigurasi pada FT Server dan DB server

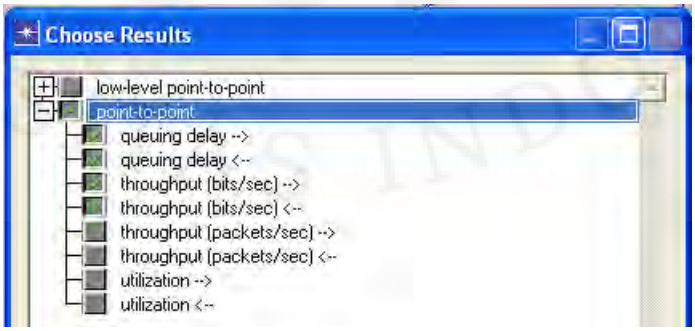
3.5 Metode Pengambilan Data

Terdapat empat skenario yang dijalankan untuk memperlihatkan performa jaringan dalam kondisi tertentu. Pengambilan data dilakukan sebanyak lima kali, yaitu untuk masing-masing skenario, lalu pengambilan data kelima dilakukan untuk menjalankan keempat skenario sekaligus agar didapat perbandingan hasilnya. Keempat skenario itu adalah:

Tabel 3.2 Skenario-skenario pengambilan data

Skenario	Karakteristik Jaringan			Parameter QoS
	VPN	TCP Window Size	WAN Link	
Skenario 1	Tidak	8 k	DS1	Aplikasi HTTP: Page Response Time, Object Response Time, Traffic Sent & Received. TCP: Delay WAN Link: Queuing delay, throughput.
Skenario 2	Ya	8 k	DS1	
Skenario 3	Ya	32 k	DS1	
Skenario 4	Ya	8 k	DS3	

Masing-masing skenario akan mencoba mengimplementasikan satu solusi untuk meningkatkan performa VPN. Untuk setiap skenario akan dilihat data statistik berupa Traffic Received (bytes/sec), Traffic Sent (bytes/sec), Page Response Time, dan Object Response Time pada aplikasi HTTP. Dan pada parameter TCP dilihat Delay (sec) dan Retransmision Count untuk melihat seberapa banyak data yang ditransmisikan ulang. Pada link PPP_DS1 yang menghubungkan Router 2 dengan Internet juga diambil statistiknya berupa *Queuing Delay* dan *Throughput (bits/sec).*

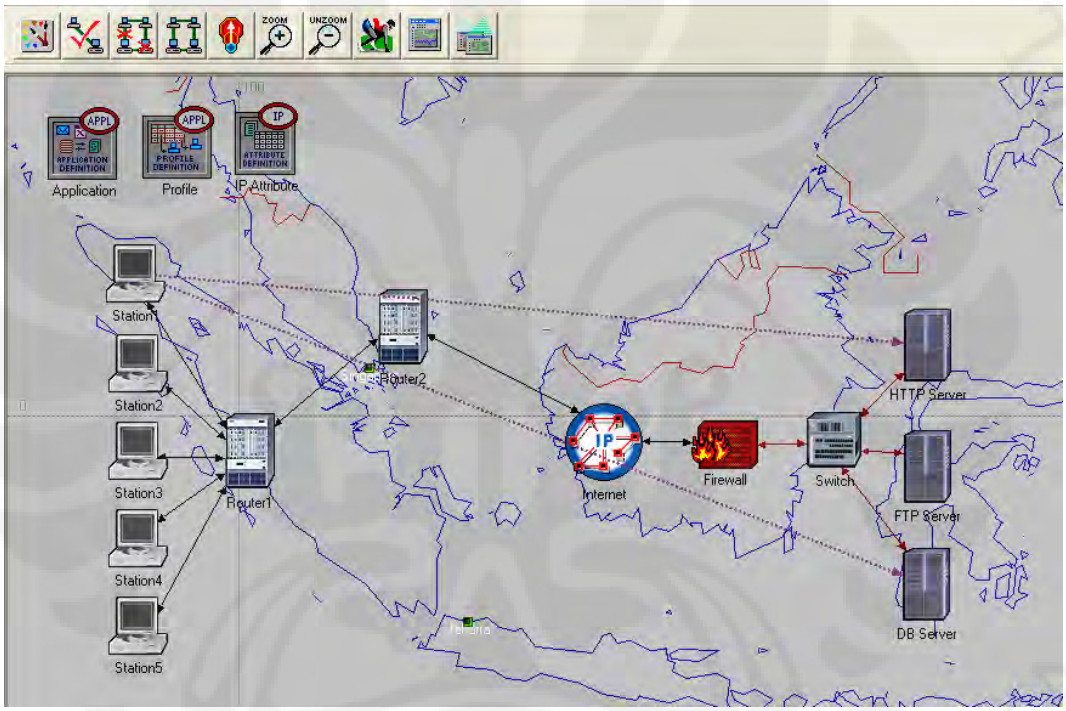


Gambar 3.9 Parameter jaringan pada link WAN

Berikut ini penjelasan lebih lanjut mengenai empat skenario untuk menguji performa jaringan VPN dengan dua solusi yang berbeda, yaitu :

1. Skenario 1 (NoVPN_DS1_8k)

Skenario ini diberi nama “NoVPN_DS1_8k”, pada skenario ini jaringan berjalan seperti jaringan WAN biasa tanpa ada tunel VPN didalamnya. “DS1” menunjukkan jenis link WAN yang digunakan, sementara “8k” menunjukkan ukuran TCP window yang dipakai, 8k (8760) merupakan ukuran default dari OPNET untuk TCP Parameter di semua node.

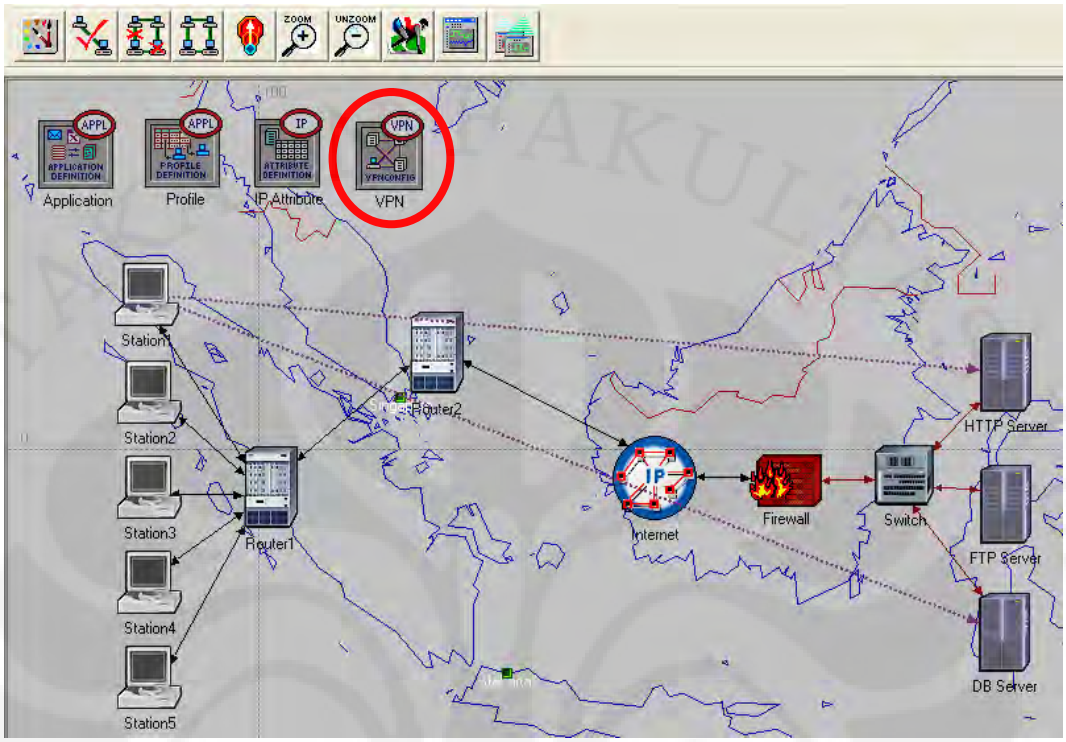


Gambar 3.10 Tampilan skenario 1

2. Skenario 2 (WithVPN_DS1_8k)

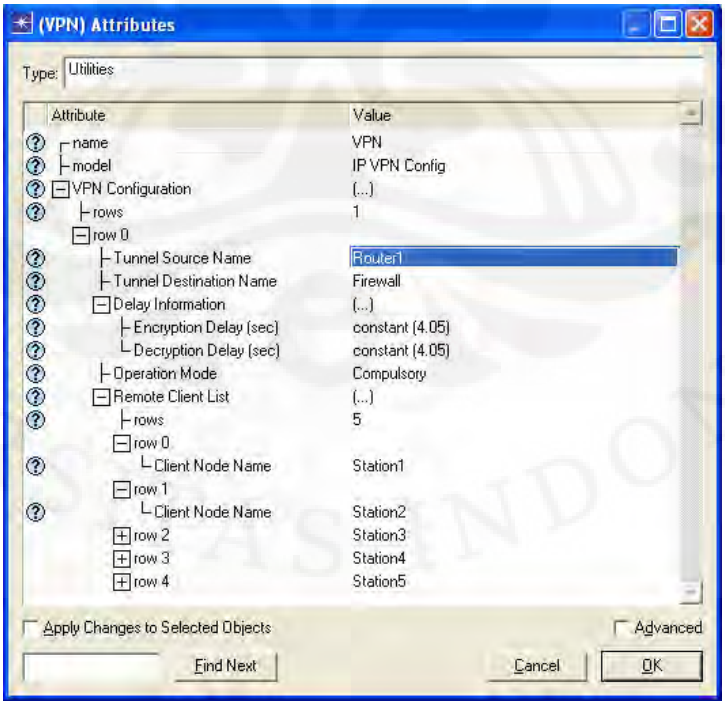
Skenario yang diberi nama “WithVPN_DS1_8k” dikonfigurasi dengan tambahan node **IP VPN Config** yang digunakan untuk mendefinisikan VPN tunel yang dibangun antara Router1 dengan Firewall. Tipe VPN tunneling yang digunakan pada skenario ini adalah tipe compulsory, dimana tunel dibangun dengan Router1 sebagai titik awal tunel dan firewall sebagai titik ujung tunel. Saluran ini merepresentasikan hubungan privat virtual antara jaringan Router 1

yang berisi 5 workstation dengan jaringan pada firewall yang merupakan *Server Farm*.



Gambar 3.11 Tampilan skenario 2 dengan VPN

Node VPN dikonfigurasi dengan settingan sebagai berikut:

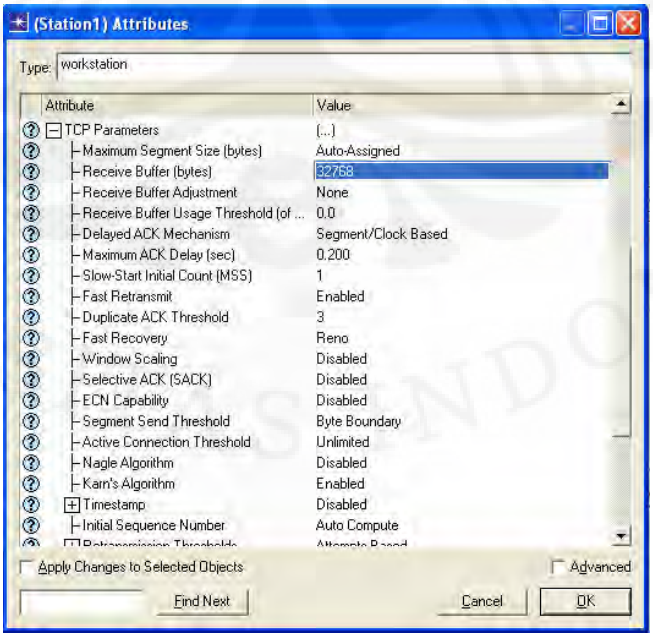


Gambar 3.12 Konfigurasi VPN

Pada baris **Delay Information**, waktu untuk proses enkripsi dan dekripsi dibuat sama untuk mengindikasikan bahwa algoritma yang digunakan untuk mengenkripsi dan men-dekripsi adalah algoritma yang sama, pada VPN kita bisa saja menggunakan algoritma yang berbeda untuk proses eknripsi dan dekripsinya. **Remote Client List** mendefinisikan node mana saja yang dapat mengakses tunel ini, node yang tidak diefinisikan pada list ini akan tetap menggunakan jalur biasa (bukan tunel VPN) meskipun node tersebut secara langsung terhubung dengan Router1 dan router ini menjadi satu-satunya jalan untuk akses jaringan, tunel hanya terbuka bagi client yang terdaftar. Pada skenario 1 dan 2 dipasang sebuah jalur pengiriman paket data ICMP atau *ping*. Station1 mengirim request ping ke server dan kemudian jejak perjalanan paket ICMP ini direkam untuk melihat proses perpindahannya, tujuannya adalah melihat perbedaan jalur pengiriman antara jaringan WAN biasa tanpa tunneling dengan jaringan yang memiliki tunel VPN.

3. Skenario 3 (WithVPN_DS1_32k)

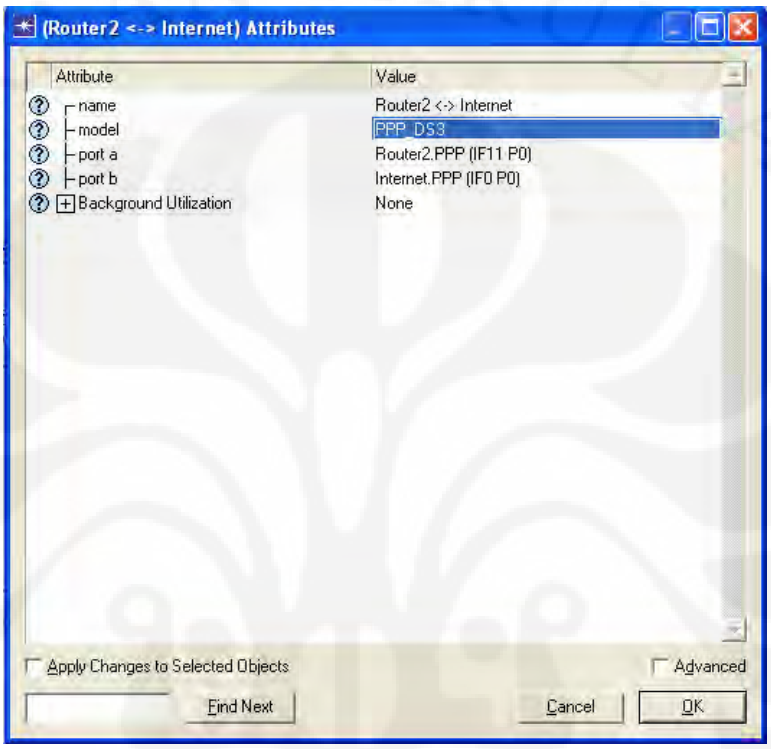
Skenario ini mencoba memberi solusi untuk mengurangi delay yang terdapat pada jaringan dengan tunel VPN dengan cara menambah ukuran TCP Window dari 8760 (8k) menjadi 32768 (32k). Setiap workstation dan server dikonfigurasi ulang dengan merubah ukutan TCP Window pada baris **TCP Parameter** > **Receive Buffer (bytes)** menjadi 32768.



Gambar 3.13 Konfigurasi pada station dan server untuk merubah TCP Window

4. Skenario 4 (WithVPN_DS3_8k)

Skenario ini menguji solusi untuk mengurangi delay VPN dengan mengganti WAN link antara Router2 dengan Internet. Link diup-grade dari DS1(kecepatan 1.544Mbps) ke DS3 (kecepatan 44.746Mbps), tapi ukuran TCP Window tidak dirubah, dibiarkan tetap dengan ukuran defaultnya yaitu 8k.



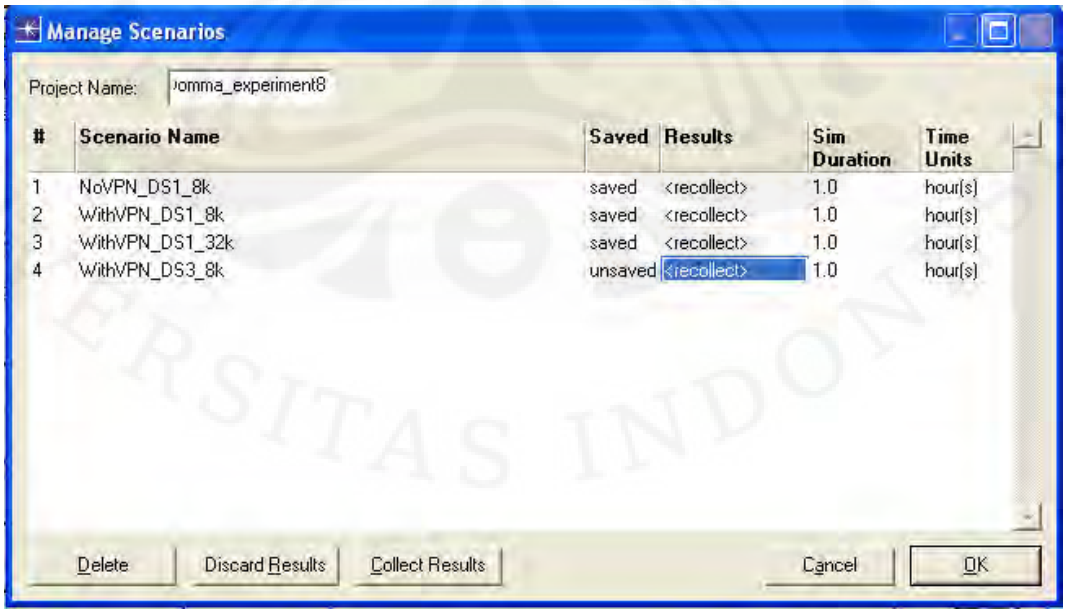
Gambar 3.14 Konfigurasi untuk merubah jenis WAN Link

BAB 4
SIMULASI DAN ANALISA

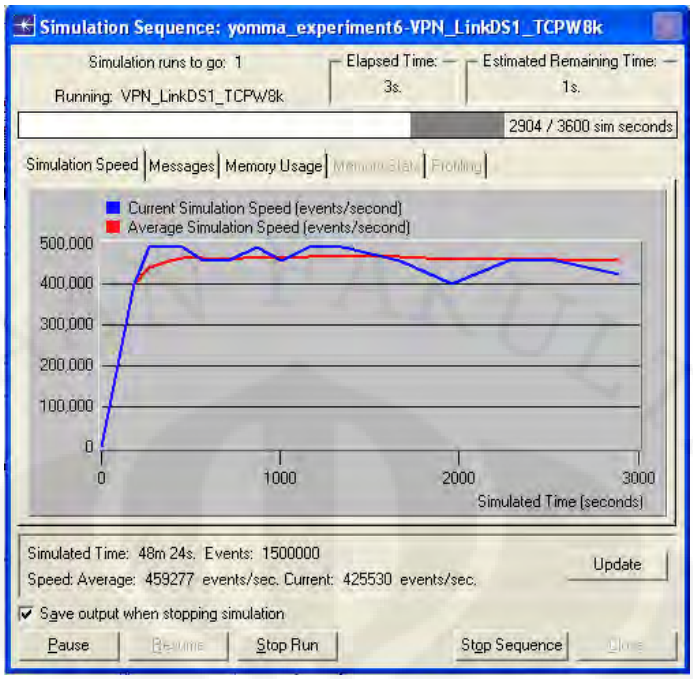
Simulasi keempat skenario yang telah dijelaskan pada bagian sebelumnya dijalankan untuk kemudian dilihat dan dianalisa, solusi yang mana yang kira-kira paling baik untuk diterapkan agar kita dapat mengimplementasikan VPN dengan delay yang lebih kecil sehingga menaikkan performa aplikasi yang berjalan pada jaringan VPN. Bagian berikut ini membahas mengenai proses simulasi jaringan dan hasil yang didapatkan.

4.1 Simulasi Jaringan Pada OPNET IT Guru

Setelah semua skenario disiapkan, kita perlu menjalankan simulasinya. Karena kita memerlukan perbandingan hasil antar skenario yang berbeda ini, maka simulasi harus dijalankan secara kolektif bersama-sama. Langkah-langkahnya adalah, klik **Scenario** pada *Toolbar* bagian atas lalu klik **Manage Scenarios**. Setelah sebuah window keluar, pada bagian **Results** pilih ***"collect"*** atau jika tidak ada bisa pilih ***"recollect"*** lalu klik **OK**. Simulasi dijalankan selama satu jam sesuai dengan durasi default dari OPNET. Lamanya simulasi tergantung spesifikasi komputer yang menjalankan simulasi OPNET.



Gambar 4.1 Inisiasi sebelum simulasi dijalankan



Gambar 4.2 Tampilan ketika simulasi sedang berjalan

4.2 Hasil Pengukuran

Output dari simulasi ini berupa grafik-grafik yang menyatakan nilai statistik dari parameter tertentu. Rangkuman hasil penelitian dapat dilihat pada tabel berikut ini:

Tabel 4.1 Hasil perbandingan skenario 1 s/d 4

Skenario	Parameter			
	Page Response Time HTTP	**Packet Loss**	**Delay TCP**	**Queuing Delay WAN Link**
Skenario 1	4.4s	6.30%	1.67s	3.9ms
Skenario 2	5.1s	2.50%	1.80s	4.4ms
Skenario 3	5.0s	1.13%	1.78s	4.2ms
Skenario 4	3.5s	0.25%	1.64s	0.2ms

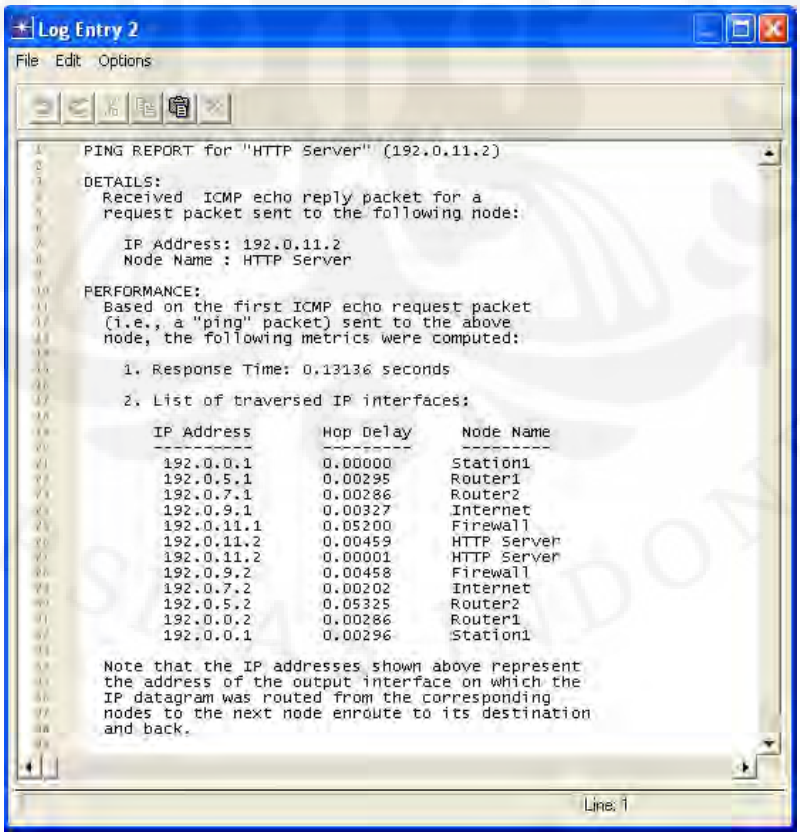
Bagian ini menjelaskan hari pengukuran lebih lanjut dari masing-masing skenario, perbandingan hasil akan dilakukan pada bagian berikutnya. Dibawah ini hasil tampilan hasil detail dari masing-masing skenario :

1. Skenario pertama, ***NoVPN_DS1_8k***

   Sebelum melihat hasil grafik skenario pertama, kita lihat dahulu jejak trafik (Record Route) pada scenario ini yang tidak menggunakan VPN. Station1 mengirimkan paket ICMP ke HTTP Server dan DB Server. Hasilnya seperti terlihat pada gambar, paket ICMP mengalir dari Station1 ke Router1, lalu diteruskan ke Router2, dan masuk kedalam jaringan Internet, selanjutnya paket dikirim ke Firewall sebelum akhirnya sampai di HTTP server. Paket balasan dikirim dengan melalui jalur yang sama namun urutan node yang terbalik.

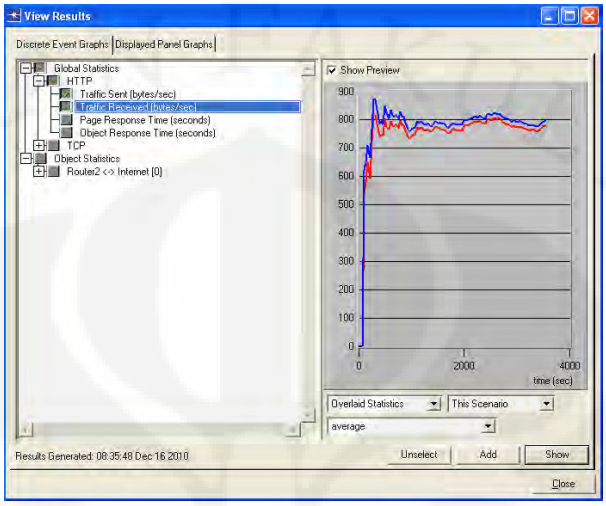
   Station1 > Router1 > Router2 > Internet > Firewall > HTTP Server

   HTTP Server > Firewall > Internet > Router2 > Router1 > Station1



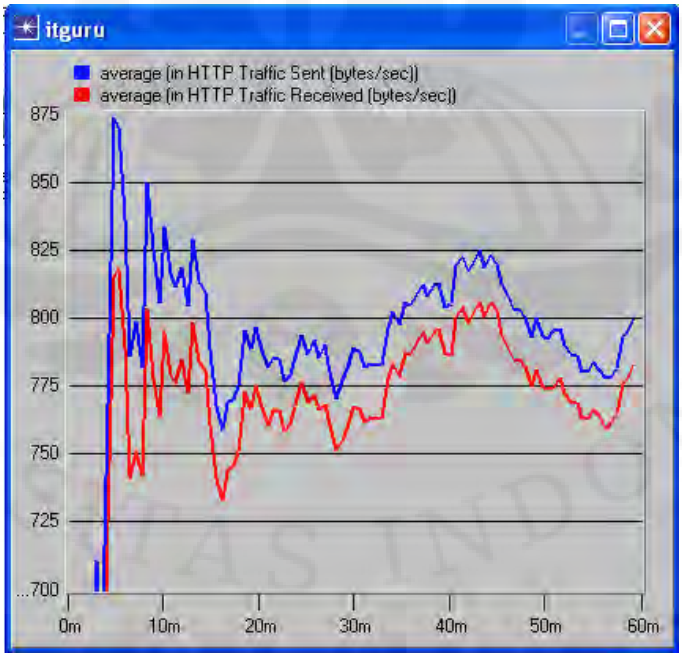
Gambar 4.3 Ping Report pada skenario tanpa VPN

Ping report ini sekaligus membuktikan bahwa perangkat-perangkat dalam jaringan sudah saling terkoneksi dengan baik. Selanjutnya kita perlu melihat seberapa besar packet loss dari aplikasi HTTP yang terjadi pada jaringan tanpa VPN ini, hasilnya terlihat pada grafik dibawah ini:



Gambar 4.4 Grafik perbandingan Traffic Sent dengan Traffic Received

Grafik menunjukkan bahwa terdapat perbedaan jumlah paket yang dikirim dengan jumlah paket yang diterima, hal ini mengindikasikan adanya paket yang hilang didalam jaringan selama masa transmisi.



Gambar 4.5 Perbandingan paket terkirim dengan paket diterima (diperbesar)

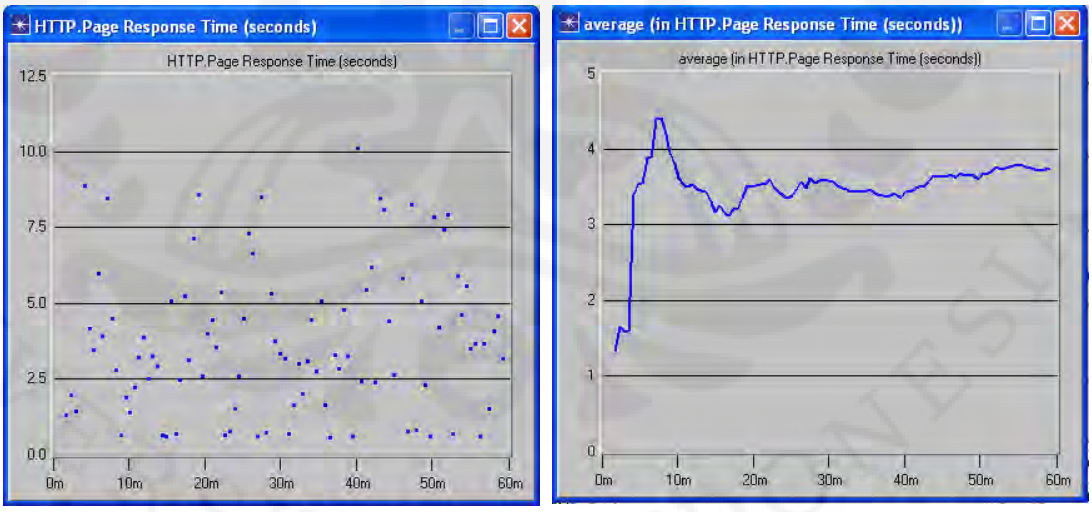
Grafik diatas menunjukkan rata-rata jumlah data yang dikirim dilihat dari titik puncak masing-masing adalah sebesar 873 bytes/sec, sementara yang diterima hanya sebesar 818 bytes/sec. Terdapat perbedaan sebesar 55 antara jumlah paket yang dikirim dengan jumlah paket yang diterima. Maka banyaknya paket yang hilang adalah sebesar:

Packet Loss :

Loss = ((Paket terkirim – Paket diterima)/Jumlah paket terkirim) x 100

Packet loss yang didapat adalah : ((873-818)/873) x 100 = 6.3%. Angka ini masih masuk kedalam kategori kondisi normal jaringan, dimana jaringan yang termasuk dalam kategori bagus adalah jaringan dengan packet loss sebesar 3% dan kategori sedang sebesar 15%. Packet loss 6.3% berada diantara dua kategori sebelumnya, maka jaringan ini masih bisa dianggap baik.

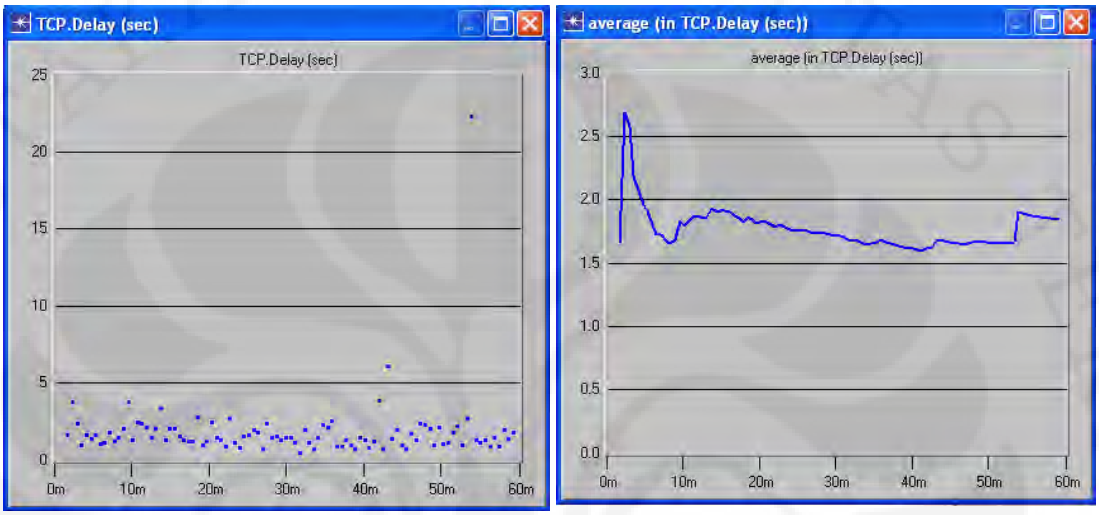
Berikutnya kita perlu melihat performa aplikasi HTTP melalui *Page Response Time*, yaitu waktu yang dibutuhkan untuk merespon input dari client.



Gambar 4.6 HTTP Page response time pada skenario 1

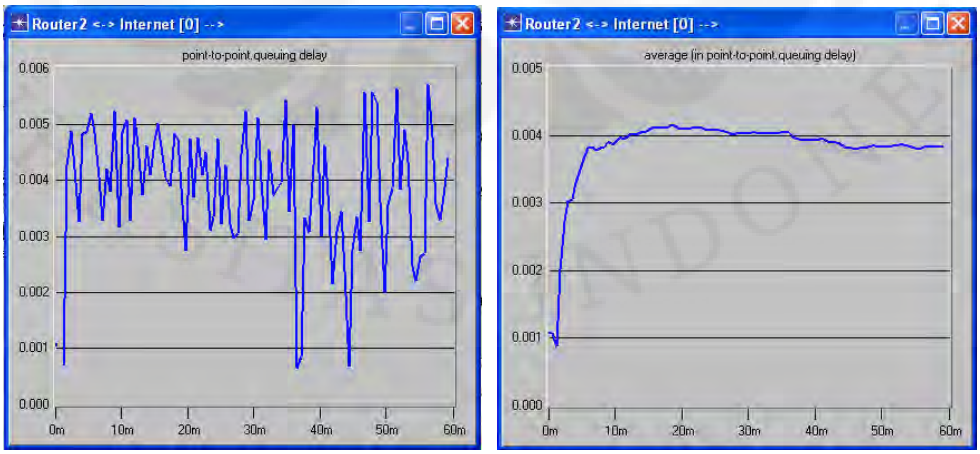
Grafik disebelah kiri (dengan titik-titik) menunjukkan waktu respon aktual dari aplikasi HTTP selama satu jam, grafik ini tidak dapat dibaca. Oleh karena itu grafik diubah dalam bentuk rata-rata agar dapat ditaksir nilainya,

dan grafik disebelah kanan adalah grafik dalam bentuk rata-rata yang memperlihatkan waktu respon rata-rata selama satu jam, dengan melihat rata-rata pada saat kondisi jaringan lebih stabil yaitu dari menit ke 15 sampai ke 60 maka rata-rata waktu respon-nya adalah sebesar 3.3s. Salah satu parameter Quality of Service berikutnya adalah delay, maka kita dapat melihat TCP delay, dan grafik menunjukkan hasil sebagai berikut :



Gambar 4.7 TCP Delay pada skenario 1

Grafik berikutnya menunjukkan *queuing delay* dalam jaringan, yang diukur dari delay pada link DS1 yang menghubungkan Router2 ke Internet. Link ini dipilih karena merupakan link yang paling merepresentasikan WAN Link.

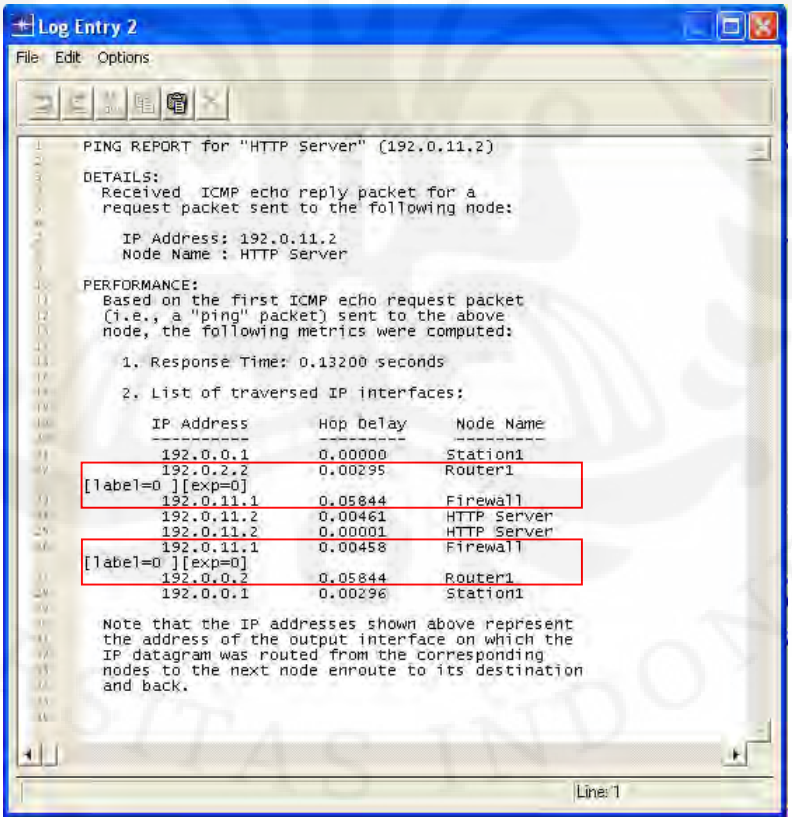


Gambar 4.8 Queuing delay pada skenario 1

Delay pada jaringan disaat jaringan sudah mencapai kestabilan yaitu sekitar 0.004 detik atau sebesar 4ms. Menurut dasar teori Quality of Service, delay ini termasuk kategori sangat bagus dimana delay yang dapat diterima untuk kebanyakan pengguna aplikasi adalah antar 0ms – 150ms.

2. Skenario kedua, ***WithVPN_DS1_8k***

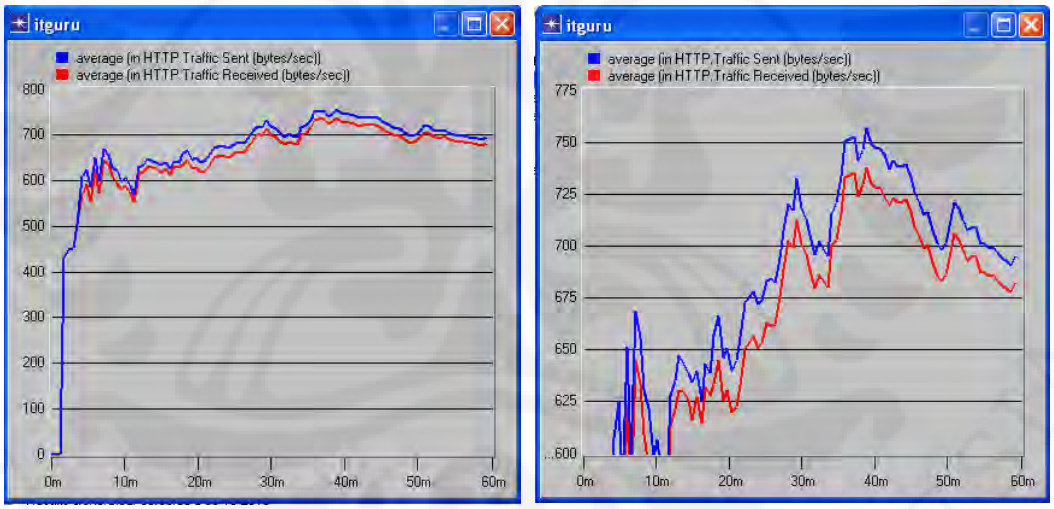
Skenario ini menerapkan implementasi tunel VPN pada WAN, untuk melihat apakah tunel VPN ini bekerja dengan baik maka kita perlu melihat ping report yang merekam jejak perjalanan paket. Seperti pada skenario pertama, pada skenario ini Station1 kembali mengirimkan paket ICMP ke Server, gambar berikut ini memperlihatkan perjalanan paket ICMP dalam jaringan dengan tunel VPN.



Gambar 4.9 Ping report pada skenario 2 dengan VPN

Ada perbedaan yang cukup mencolok terkait rute perjalanan paket pada jaringan yang menggunakan tunel VPN dengan yang tidak. Terlihat bahwa

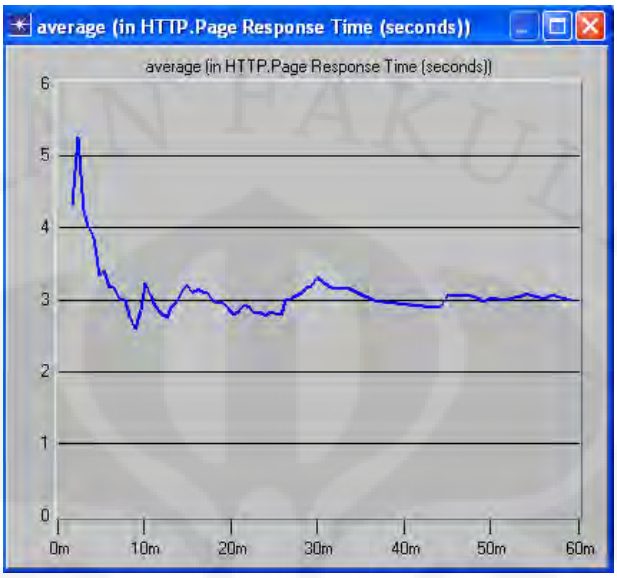
paket ICMP setelah dikirim Station1 lalu menuju ke Router1 yang menjadi gerbang awal tunel VPN langsung menuju ke node Firewall tanpa melewati Router2, dan Internet terlebih dahulu seperti yang seharusnya sesuai dengan topologi jaringan. Model VPN compulsory yang diterapkan pada skenario ini membuat paket dialirkan secara langsung dari ujung awal tunel ke ujung satunya, routing akan dilakukan dengan terlebih dahulu mengirimkan paket dari satu titik tunel VPN ketitik lainnya, baru setelah itu dikirim ke tujuan, routing ini akan tetap digunakan meskipun bukan merupakan jalur terpendek, jadi aturannya adalah paket harus terlebih dahulu melalui kedua ujung tunel baru dikirim ke tujuan. Tanda "***[label=0] [exp=0]***" menunjukkan proses enkripsi dan dekripsi yang dilakukan pada kedua ujung tunel VPN, terlihat pada ping report bahwa proses ini terulang ketika pengiriman paket kembali ke client, artinya paket yang dikirim melalui tunel VPN akan selalu berada dalam bentuk ter-enkripsi. Inilah yang membuat VPN menjadi sebuah saluran yang aman untuk mengirimkan data melalui jaringan publik seperti internet.



Gambar 4.10 Perbandingan paket terkirim dan diterima pada skenario 2

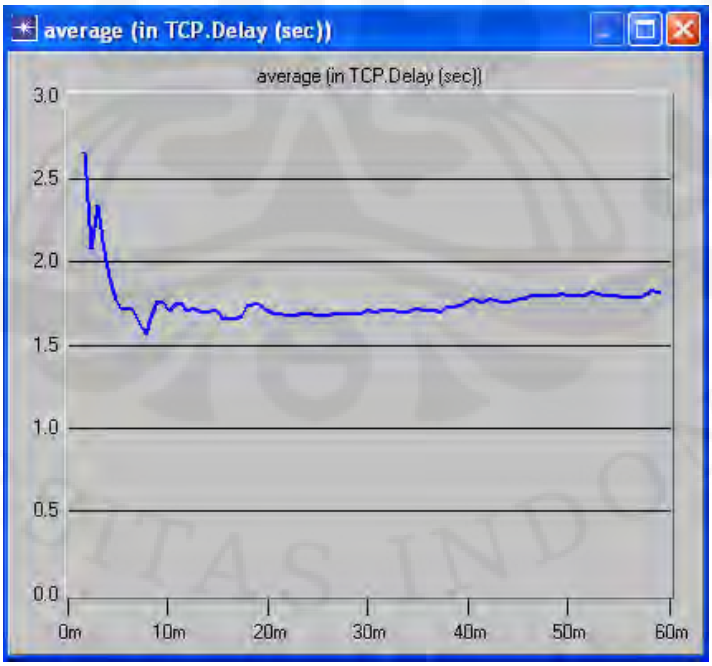
Seperti halnya karakteristik alami jaringan komputer, pada VPN pun terdapat packet loss. Hal ini terlihat didalam grafik, paket yang dikirim sebesar 757 dan paket yang diterima hanya sebesar 734, sehingga paket yang hilang sebanyak 19 dan menghasilkan persentase packet loss sebesar 2.5%.

Grafik berikutnya menunjukkan page respons time aplikasi HTTP pada jaringan dengan tunel VPN:



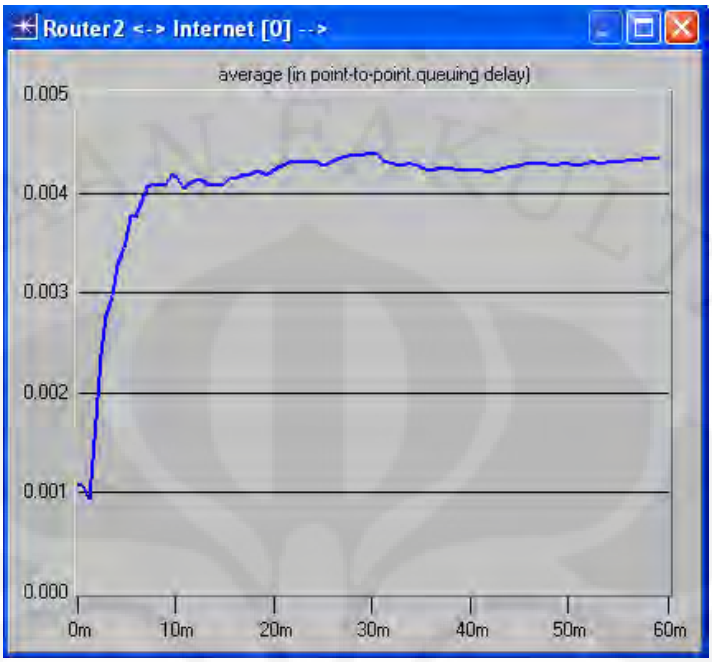
Gambar 4.11 HTTP page response time pada skenario 2

Grafik berikut memperlihatkan besaran TCP delay rata-rata sebesar 1.75 detik pada skenario kedua:



Gambar 4.12 TCP delay pada skenario 2

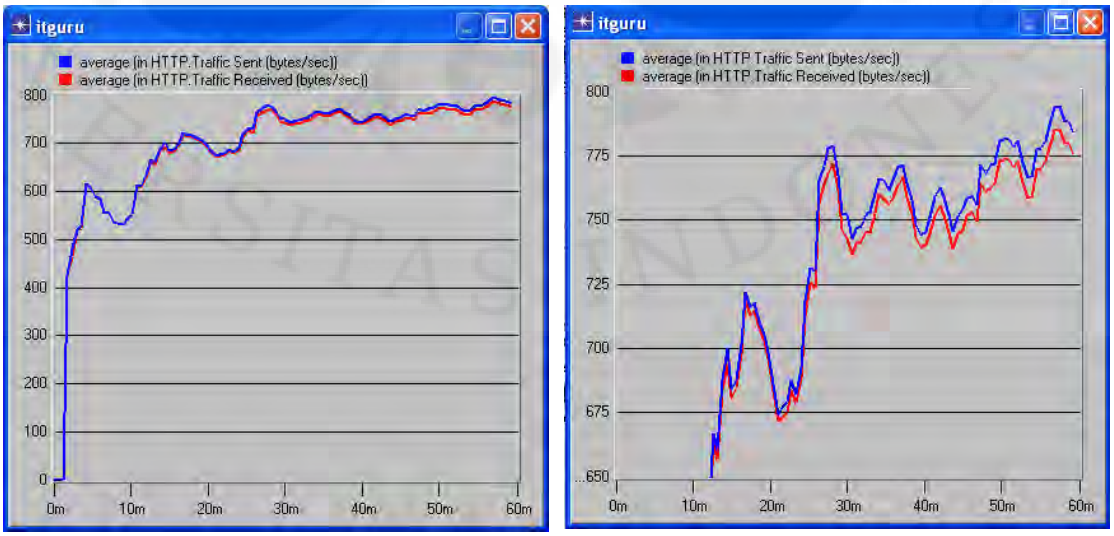
Sedangkan untuk queuing delay pada jaringan yang diukur di link DS1 Router1 ↔ Internet menunjukkan delay sebesar 0.0043 detik atau 4.3ms:



Gambar 4.13 Queuing delay pada skenario 2

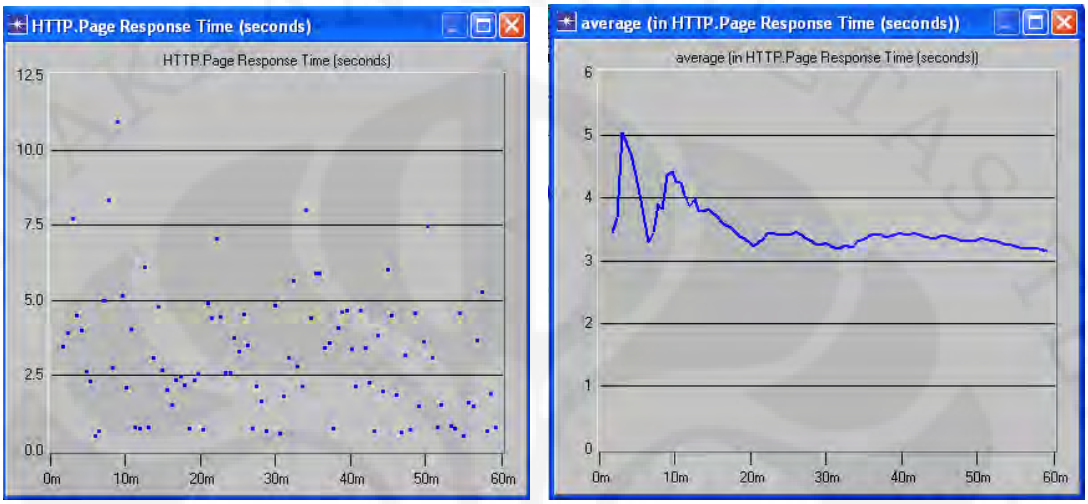
3. Skenario ketiga, ***WithVPN_DS1_32k***

Skenario ketiga hampir sama dengan skenario kedua, namun pada skenario ini delay pada jaringan diturunkan sehingga meningkatkan performa jaringan. Caranya dengan merubah ukuran TCP Window dari 8760 (8k) menjadi 32768 (32k) sehingga paket yang dikirimkan dalam satu satuan waktu akan lebih besar dan mempercepat proses pengiriman data. Grafik berikut memperlihatkan perbandingan paket yang dikirim dan diterima:



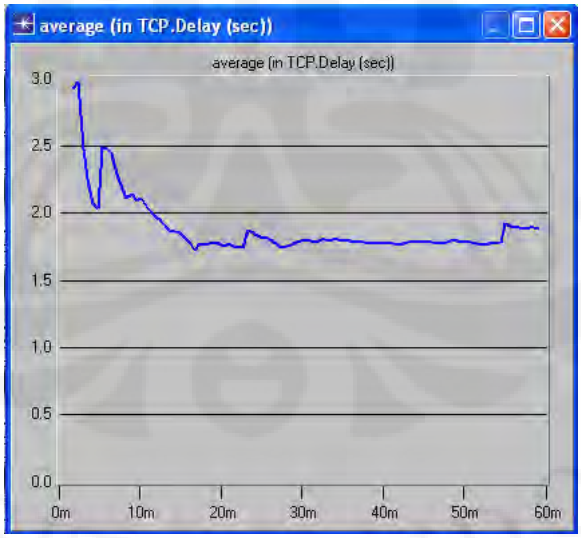
Gambar 4.14 Perbandingan paket terkirim dan diterima pada skenario 3

Jumlah rata-rata paket terkirim adalah 794 bytes/sec sementara yang diterima sebesar 785 bytes/sec menyisakan sebanyak 9 paket yang hilang, sehingga packet losss-nya adalah sebesar 1.13%. Berikutnya grafik waktu respon HTTP:



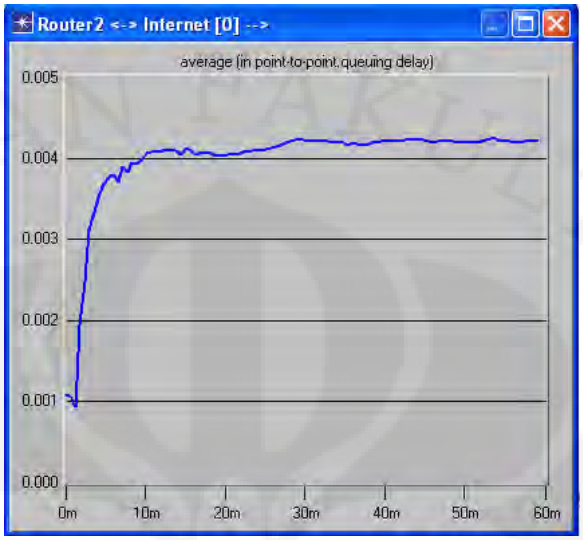
Gambar 4.15 HTTP Page Response Time pada skenario 3

Delay TCP menunjukkan angka sekitar 1.7 detik:



Gambar 4.16 TCP delay pada skenario 3

Grafiknya menunjukkan delay pada jaringan yang diukur pada link WAN antara Router2 dengan Internet menunjukkan angka kira-kira sekitar 0.0041 detik atau 4.1ms:

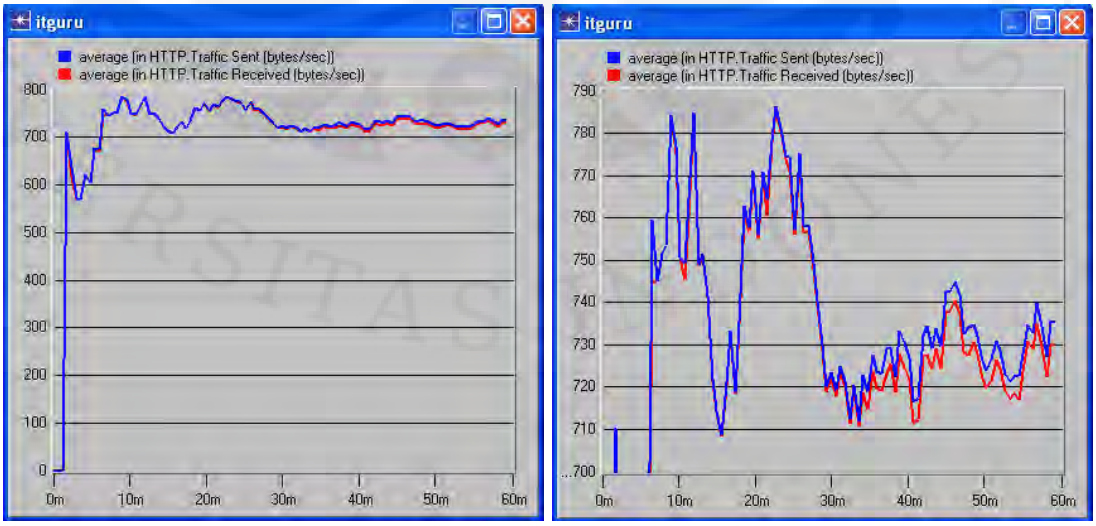


Gambar 4.17 Queuing delay pada skenario 3

4. Skenario keempat, ***WithVPN_DS3_8k***

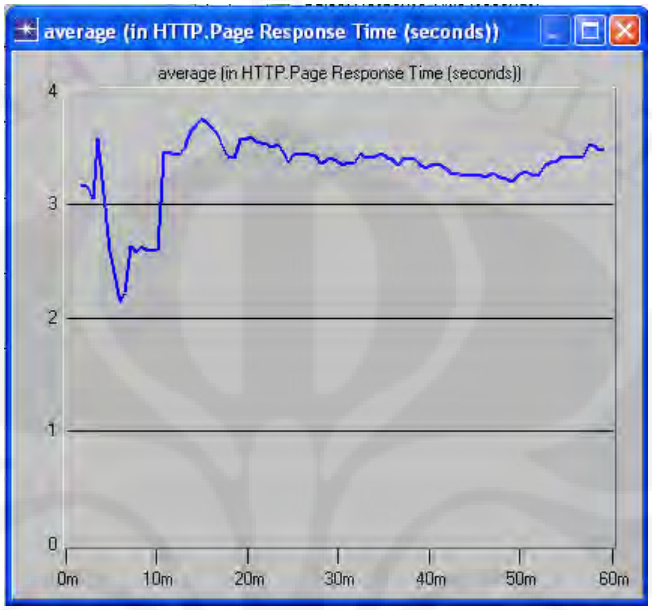
Pada skenario ini link WAN yang menghubungkan Router2 dengan Internet diup-grade dari DS1(kecepatan 1.544Mbps) menjadi DS3 (kecepatan 44.746Mbps).

Grafik berikut memperlihatkan perbedaan jumlah trafik yang dikirim dengan trafik yang diterima:



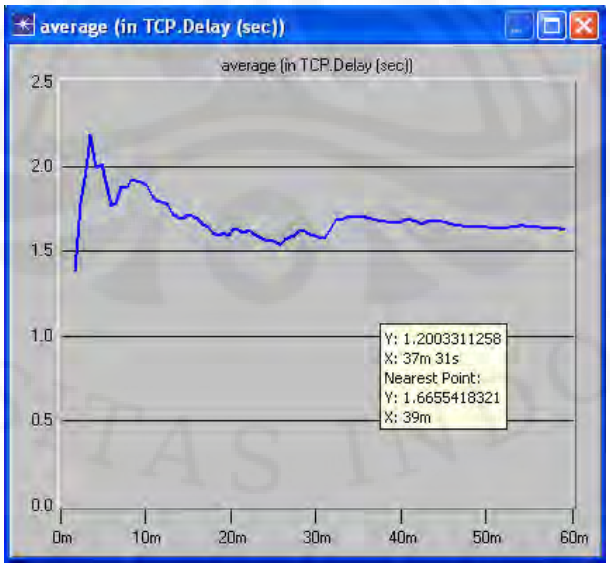
Gambar 4. 18 Perbandingan paket terkirim dan diterima pada skenario 4

Rata-rata jumlah data terkirim sekitar 786 bytes/sec, sementara data yang diterima 784 bytes/sec. Hilangnya hanya 2 byte data ini membuat packet loss pada jaringan dengan DS3 sebagai link WAN hanya sebesar 0.25%. Berikut ini adalah grafik page response time aplikasi HTTP:



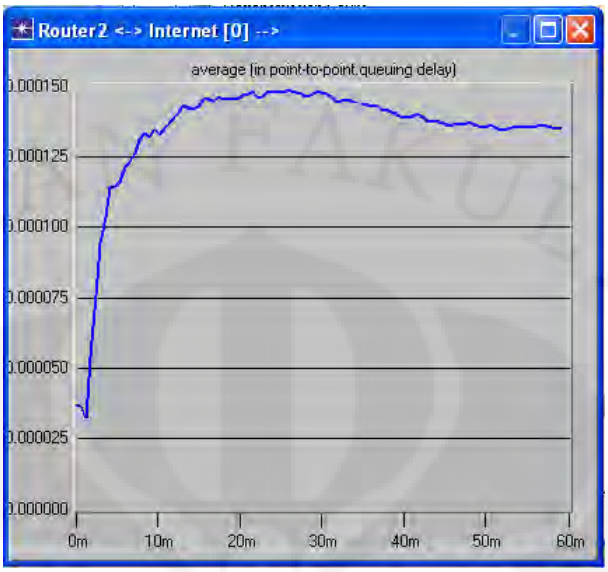
Gambar 4.19 HTTP page response time pada skenario 4

Grafik TCP delay pada jaringan VPN dengan link WAN yang diup-grade dari DS1 ke DS3:



Gambar 4.20 TCP delay pada skenario 4

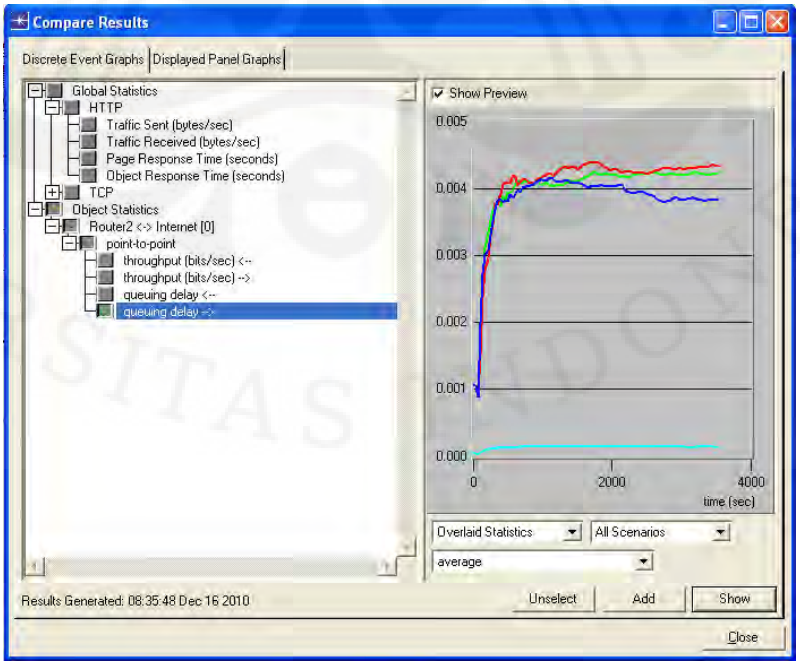
Dan grafik dibawah menunjukkan delay pada link WAN DS3 adalah sebesar 0.00013 detik atau sekitar 0.13ms:



Gambar 4.21 Hasil queuinng delay pada WAN link di skenario 4

4.3 Analisa Perbandingan Skenario 1,2,3,&4

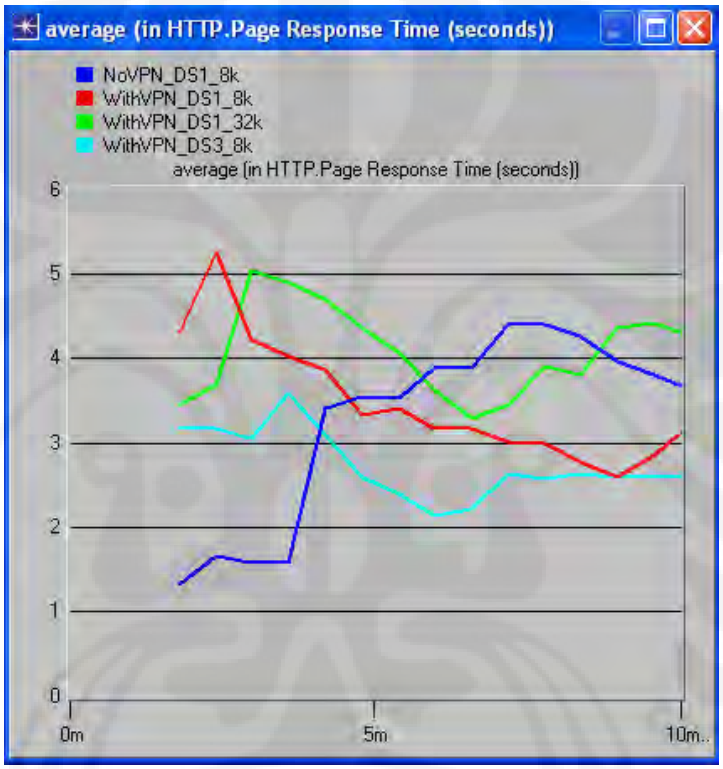
Untuk dapat membandingkan hasil simulasi keempat skenario dapat dilakukan dengan cara klik kanan pada wilayah kosong diatas grid lalu pilih Compare Result, maka kita akan dapat melihat tampilan grafik yang berisi hasil simulasi keempat skenario dengan parameter yang sama:



Gambar 4.22 Tampilan grafik perbandingan hasil keempat skenario

Grafik ini ditampilkan dengan perbedaan warna yang masing-masing warna merepresentasikan setiap skenario, dimulai dari skenario satu, maka urutan warna dari setiap skenario adalah Biru, Merah, Hijau, dan Biru muda terang. Analisa perbandingan ini akan lebih fokus pada perbedaan hasil output parameter-parameter yang memperlihatkan kualitas performa aplikasi pada jaringan VPN.

a. Parameter Page Response Time pada aplikasi HTTP

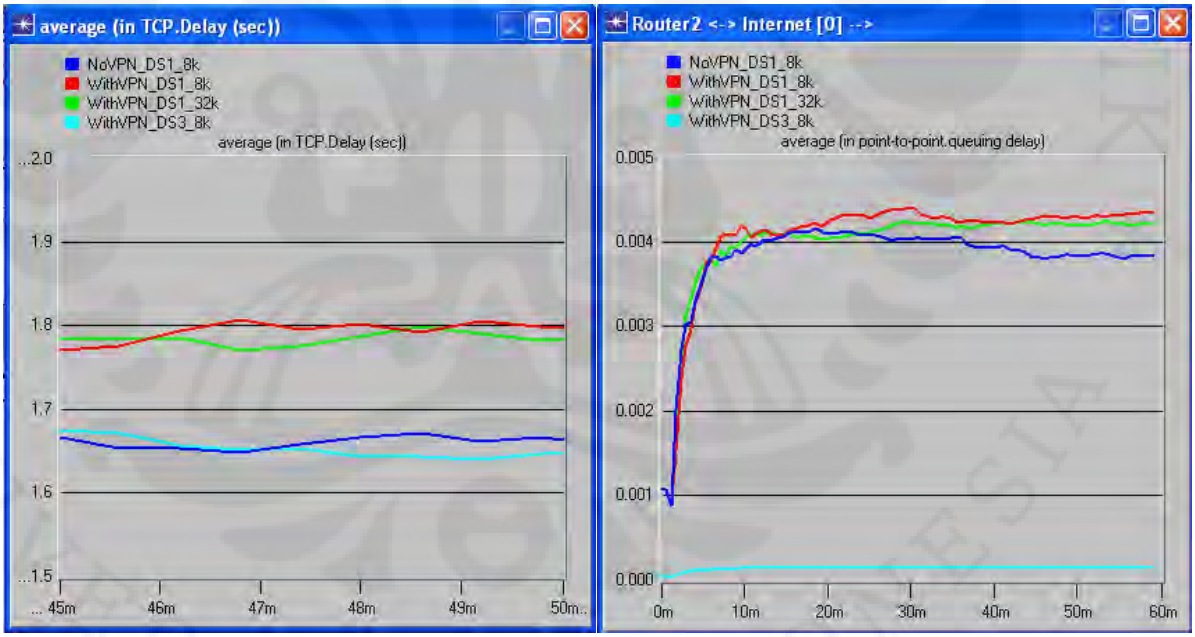


Gambar 4.23 Perbandingan HTTP Page response time

Terlihat bahwa page response time pada jaringan dengan tunel VPN lebih tinggi dibandingkan dengan jaringan tanpa VPN. Dengan melihat titik-titik puncak yang dicapai masing-masing grafik, garis berwarna merah yang merepresentasikan jaringan dengan VPN (WithVPN_DS1_8k) mencapai titik lebih tinggi dari pada grafik berwarna biru yang merepresntasikan jaringan normal tanpa VPN (NoVPN_DS1_8k). Hal ini membuktikan bahwa aplikasi Streaming Multimedia yang berjalan pada HTTP memiliki performa yang lebih baik jika tidak ditransmisikan melalui tunel VPN. Karena pada VPN

paket yang dikirim dienkripsi terlebih dahulu, dan sesampainya dibagian penerima paket di-dekripsi kembali, proses enkripsi dan dekripsi ini membutuhkan waktu sehingga menyebabkan delay yang pada akhirnya memperlambat response time pada HTTP. Namun dengan merubah ukuran TCP window menjadi lebih besar terbukti dapat menaikkan performa aplikasi seperti yang terlihat pada grafik, dari kira-kira sebesar 5.2s pada VPN dengan ukuran TCP window 8k, page response time lebih cepat menjadi 5s pada jaringan VPN dengan ukuran TCP window 32k (garis berwarna hijau). Pilihan solusi dengan meng-upgrade link WAN dari DS1 ke DS pada skenario keempat terbukti menajdi solusi paling ampuh karena meningkatkan perfoma aplikasi dengan cukup signifikan, terlihat bahwa grafik berwarna biru muda terang yang merepresentasikan skenario dengan solusi link DS3 mencapai tingkat page response time paling baik.

b. Parameter Delay



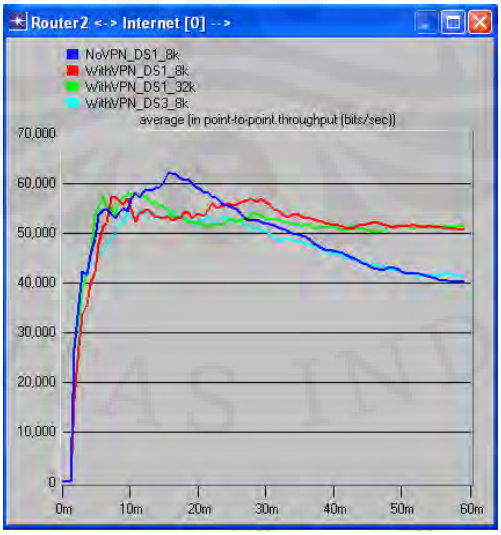
Gambar 4.24 Perbandingan TCP delay (kiri), dan queuing delay (kanan)

Grafik sebelah kiri menunjukkan TCP delay pada masing-masing skenario, dan grafik disebelah kanan menunjukkan delay jaringan yang diukur pada WAN link yang menghubungkan Router2 ↔ Internet. Delay pada jaringan dengan tunel VPN lebih tinggi karena proses enkripsi-dekripsi yang

dilakukan protokol VPN pada paket-paket yang ditransmisikan melalui tunelnya. Namun dengan merubah ukuran TCP Window ternyata dapat menurunkan delay, hal ini dikarenakan ukuran TCP Window mengindikasikan banyaknya paket yang ditransmiskan dalam satu satuan waktu, sehingga semakin besar ukuran TCP window maka akan semakin banyak data yang dibawa dan mempercepat waktu transmisi.

c. Parameter Packet Loss

Jika merujuk kembali pada hasil perhitungan persentase paket yang hilang pada setiap skenario, kita mendapatkan hasil yang menarik dari simulasi ini dimana jumlah packet loss menunjukkan kecenderungan menurun pada setiap solusi yang ditawarkan. Paket loss pada VPN dengan ukuran TCP Window 8k adalah sebesar 2.5%, jumlah ini menurun jika ukuran TCP window diganti menjadi 32k, yaitu menjadi sebesar 1.13%. Analisanya adalah semakin besar ukuran paket yang dibawa dalam satu satuan waktu maka akan semakin mempersingkat *sequence* atau rentetan pengiriman data. Hal ini juga mengurangi kejadian paket yang hilang, karena semakin banyak sequence pengiriman data maka semakin besar peluang terjadinya paket yang hilang didalam jaringan. Solusi dengan mengga nti link WAN menunjukkan hasil yang lebih mencengangkan lagi, yaitu paket loss tereduksi hingga hanya sebesar 0.25%.



Gambar 4.25 Perbandingan throughput

BAB 5
KESIMPULAN

Dari hasil penelitian pada skripsi ini, didapat beberapa kesimpulan sebagai berikut:

1. Penggunaan Virtual Private Network (VPN) pada jaringan WAN dapat meningkatkan delay pada jaringan, hal ini terlihat dari penambahan besar delay TCP dari 1.67s jika tidak menggunakan VPN menjadi 1.80s jika menggunakan VPN. Hal ini dikarenakan oleh proses enkripsi-dekripsi pada tunel VPN.

2. TCP Windowing dapat menjadi salah satu solusi yang cukup baik untuk mengatasi masalah VPN, hal ini terlihat dari data yang didapat dimana dengan menaikkan ukuran TCP window dari 8k menjadi 32k, dapat menurunkan delay dari 1.80s menjadi 1.78s, dan performa aplikasi streaming pada HTTP menunjukkan perbaikan yaitu dari nilai page response time sebesar 5.1s menjadi 5.0s. Queuing delay pada link WAN pun menunjukkan penurunan delay yaitu dari 4.4ms menjadi 4.2ms, penurunan 0.2ms merupakan perbaikan yang cukup signifikan.

3. Solusi mengganti link WAN menunjukkan performa yang jauh lebih baik dari skenario lainnya, page response time HTTP menurun menjadi 3.5s, dan delay TCP menurun menjadi 1.64s. Bahkan terjadi penurunan drastis pada queuing delay pada link WAN menjadi sebesar 0.2 ms. Solusi ini meskipun menunjukkan hasil yang sangat baik namun tentu saja juga sangat mahal karena harus mengganti infrastruktur jaringan.

4. Peningkatan performa jaringan VPN dengan solusi TCP windowing memperlihatkan perbedaan yang cukup berarti, meskipun peningkatannya tidak sebesar yang terjadi pada jaringan dengan link WAN DS3, namun solusi TCP windowing dapat menjadi pilihan karena tidak perlu merombak infrastruktur jaringan sehingga lebih praktis.

DAFTAR REFERENSI

[1] CCNA 4 Exploration; Accessing the WAN Chapter 1 (Course Material), Cisco System. (n.d.*). Sevices in a Converged WAN*. USA. 2009.

[2] Cisco Internetworking Technology Handbook (Chapter 3). *Introduction to WAN Technologies*. Cisco System. http://docwiki.cisco.com/wiki/Internetworking_Technology_Handbook, diakses November 2010.

[3] CCNA 4 Exploration; Accessing the WAN Chapter 2 (Course Material), Cisco System. (n.d.). *Point-to-Point Protocool (PPP).* USA. 2009.

[4] McQuerry, Steve (2003). *CCNA Self-Study: Interconnecting Cisco Network Devices (ICND)*, Second Edition. Cisco Press.

[5] CCNA 4 Exploration; Accessing the WAN Chapter 6 (Course Material), Cisco System. (n.d.). *Providing Teleworker Services*. USA. 2009.

[6] CCNA 4 Exploration; Accessing the WAN Chapter 3 (Course Material), Cisco System. (n.d.). *Frame Relay*. USA. 2009.

[7] Suaditya, I Nyoman. dkk. *Cara Kerja TCP/IP*. Universitas Udayana.2009.

[8] *Pengantar TCP/IP*. http://www.oocities.com/wilianto_jh/, diakses November 2010.

[9] Kozierok, Charles M (2005). *The TCP/IP Guide* (Vers. 3.0). http://www.TCPIPGuide.com, diakses november 2010

[10] Desai, Anil (2001). *Private and Secure : The VPN Solution*. http://mcpmag.com/articles/2001/03/29/private-and-secure-the-vpn-solution.aspx, diakses November 2010.

[11] *VPN Tunneling Protocol*. http://technet.microsoft.com/en-us/library/cc771298%28WS.10%29.aspx, diakses November 2010.

[12] *Quality of Service (QoS) Dan Pengukurannya*. 2010

[13] Peterson, Larry L. Davie Bruce S. *Computer Networks, A System Approach*. Edisi ketiga. 2003

[14] Bourdoucen, H. dkk. *Impact of Implementing VPN to Secure Wireless LAN*. International Journal of Computer, Information, and System Science. 2009.

[15] Wijaya, I Putu Prima. dkk. *Flow Control dan Error Pada Data Link Control*. 2010.

[16] *Virtual Private Network (VPN) sebagai alternatif Komunikasi Data Pada Jaringan Skala Luas*. http://kambing.ui.ac.id/onnopurbo/library/library-ref-ind/ref-ind-3/network/VPN_jurnal.pdf, diakses November 2010.

LAMPIRAN

OPNET Manual Guide

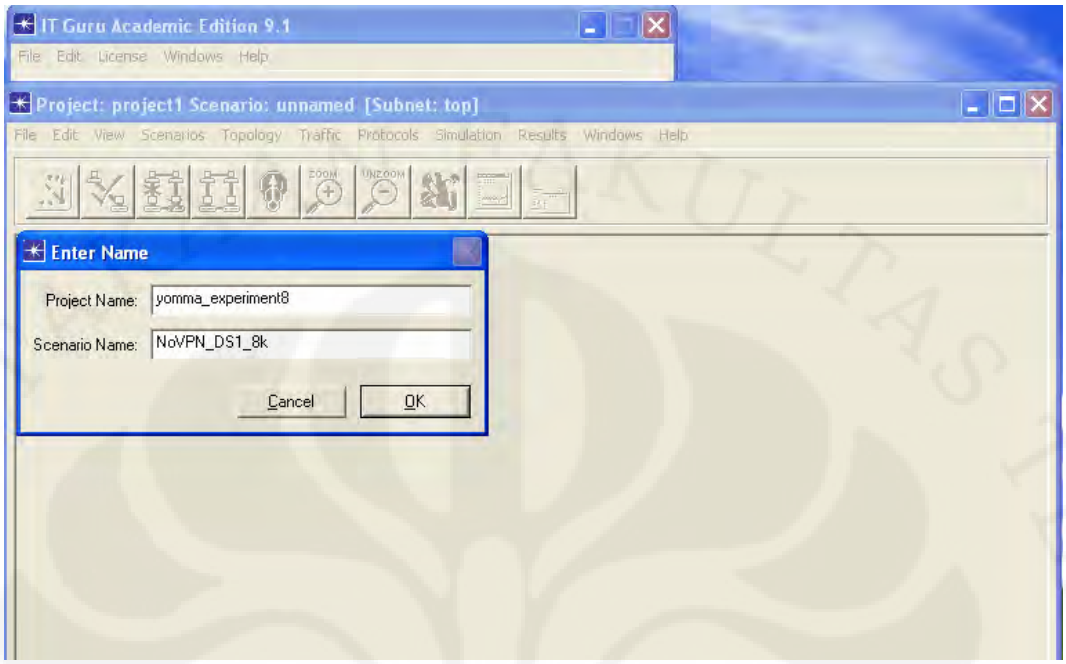
Setelah mendownload OPNET IT Guru Academic Edition 9.1 dari situs resmi software tersebut (www.opnet.com), lalu lakukan instalasi program dikomputer, proses ini akan memakan waktu beberapa menit. Dalam tahap instalasi kita akan diminta untuk memasukkan *Lisence Code* yang didapat dari situs resmi, sebelum bisa mendownload software ini, kita harus melakukan registrasi terlebih dahulu dan kemudian akan mendapat *username* dan *password* untuk mengakses situs resmi, dan untuk mendapatkan kode lisensi.

Langkah pertama yang harus dilakukan yaitu membuka aplikasi OPNET IT Guru Academic Edition 9.1 :

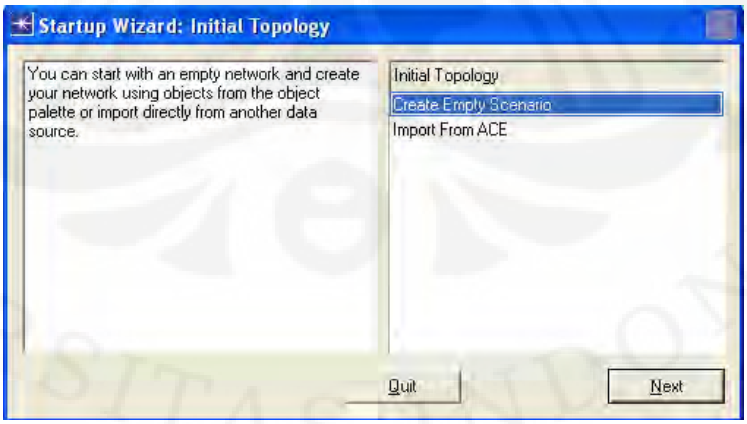
Gambar 3.2 Tampilan Awal OPNET IT Guru

Lalu pilih : **File** > **New**, dan akan muncul pilihan seperti berikut ini, lalu tekan **OK**

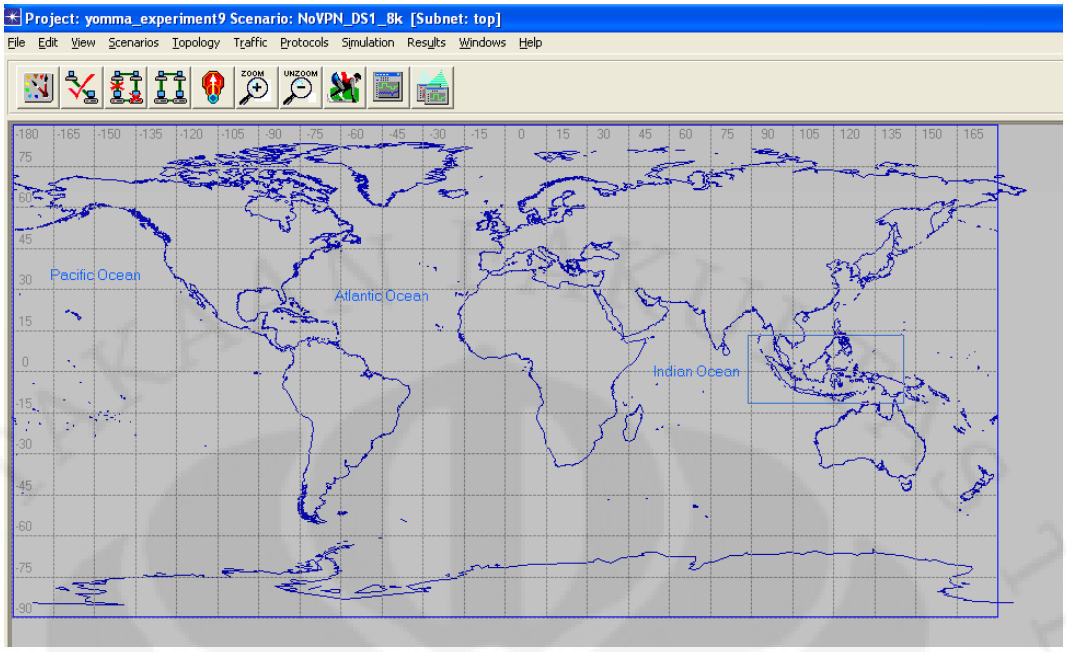
Setelah itu muncul window baru yang meminta kita untuk mengisikan nama project dan skenario seperti dibawah ini :



Berhubung untuk skripsi kali ini, yang digunakan adalah percobaan yang ke-delapan maka saya beri nama project: *yomma_experiment8* dengan skenario: *NoVPN_DS1_8k*, nama skenario ini akan dijelaskan lebih lanjut pada bagian berikutnya. Setelah semua terisi lalu tekan **OK**. Kemudian pada window *Start-Up Wizard : Initial Topology* yang muncul berikutnya pilih **Create Empty Scenario**

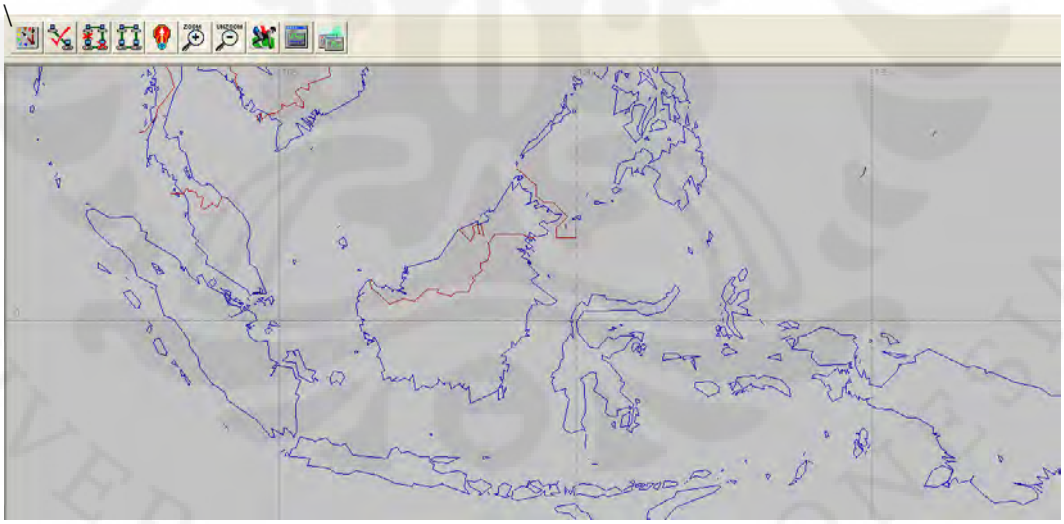


Lalu klik **Next** beberapa kali hingga window akhir klik **OK**. Akan muncul sebuah *workspace* berupa grid yang menunjukkan peta dunia.



Gunakan tombol zoom (

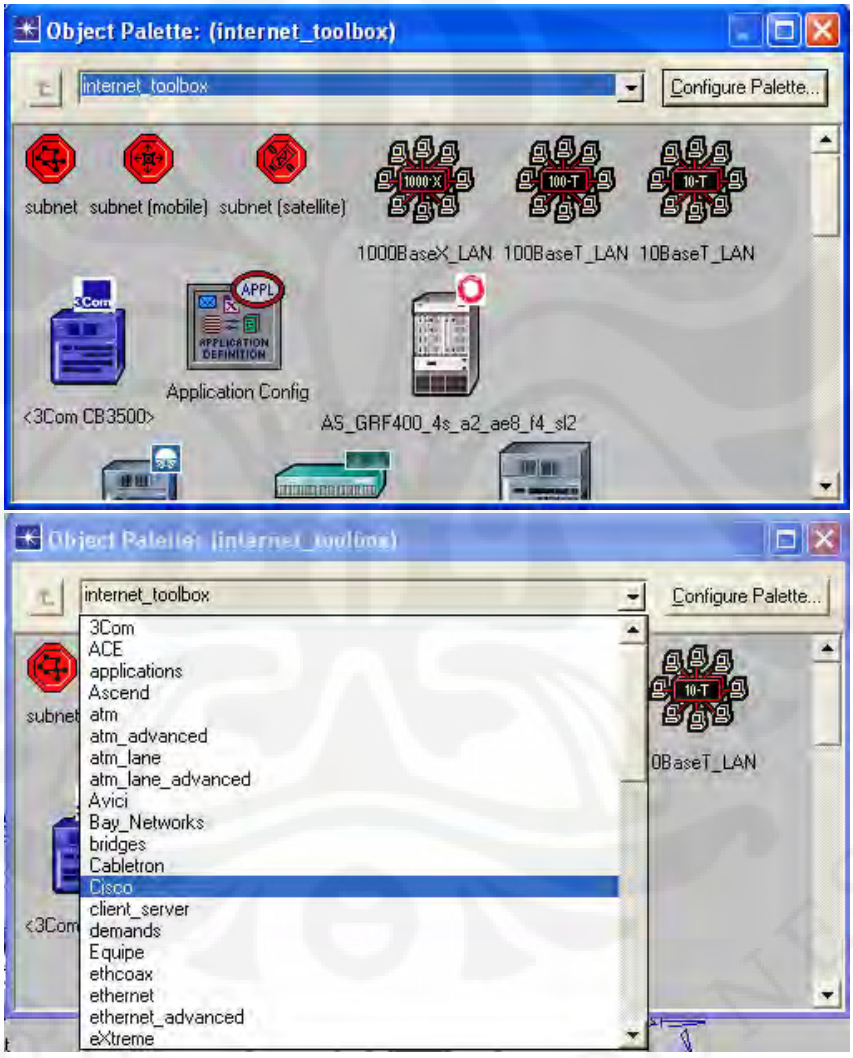
) untuk menyorot dan memperbesar wilayah Indonesia, selanjutnya tampilan workspace akan berubah menunjukkan peta wilayah Indonesia seperti berikut ini:



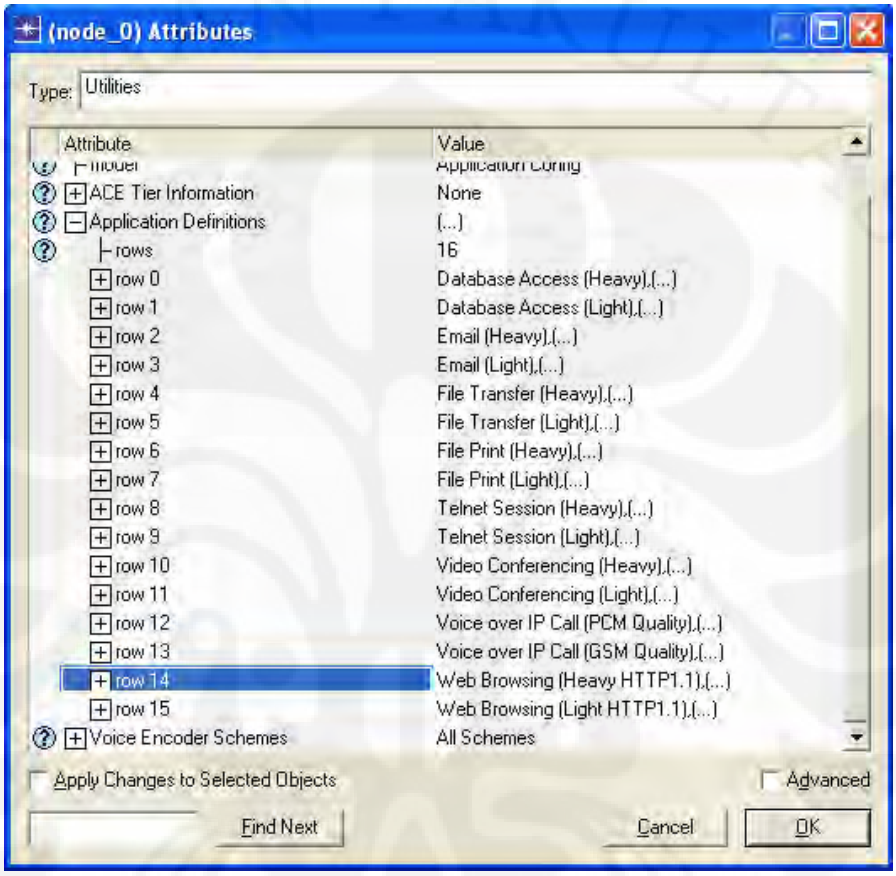
Langkah selanjutnya adalah kita perlu menebar perangkat-perangkat jaringan yang sesuai dengan rancangan topologi sebelumnya diatas grid. Objek-objek yang merepresentasikan perangkat-perangkat jaringan pada OPNET IT

Guru tersedia pada *Palette Object* yang dapat diakses dengan menekan tombol ini ().

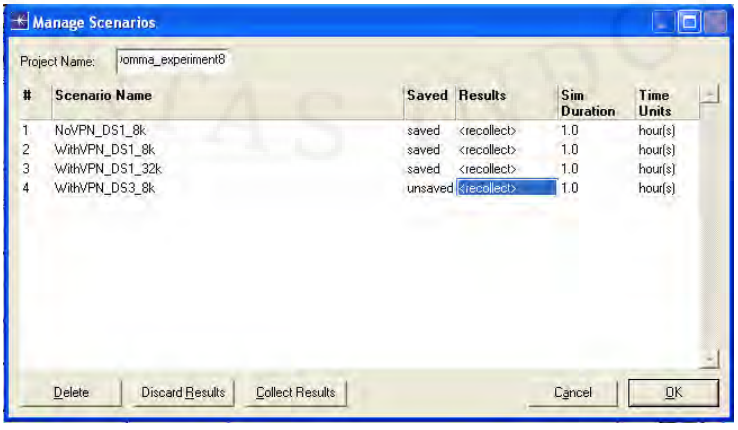
Kemudian akan muncul sebuah window yang berisikan begitu banyak perangkat-perangkat jaringan, perangkat ini diklasifikasikan berdasarkan beberapa aspek mencakup ; fungsionalitas, teknologi, dan vendor. Berikut ini tampilan Object Palette:



Hal yang pertama yang perlu dikonfigurasi adalah aplikasi yang dijalankan pada jaringan ini. Klik kanan pada node **Application** lalu pilih **Edit Attribute**, kemudian akan muncul window baru dan pada **Application Definition** pilih **Default**, hal ini akan menyebabkan OPNET secara otomatis membuat 16 jenis aplikasi.



Langkah-langkah untuk menjalankan simulasi dengan beberapa skenario adalah, klik **Scenario** pada *Toolbar* bagian atas lalu klik **Manage Scenarios**. Setelah sebuah window keluar, pada bagian **Results** pilih ***"collect"*** atau jika tidak ada bisa pilih ***"recollect"*** lalu klik **OK**.



Simulasi dijalankan selama satu jam sesuai dengan durasi default dari OPNET. Lamanya simulasi tergantung spesifikasi komputer yang menjalankan simulasi OPNET.