ANALISA SISTEM KEAMANAN JARINGAN YANG BERARSITEKTUR MPLS (MULTI PROTOCOL LABEL SWITCHING)

TUGAS AKHIR

Diajukan Sebagai Salah Satu Syarat
Untuk Memperoleh Gelar Sarjana
Jurusan Teknik Informatika

Oleh :

Nama : Aan Kurniawan

No. Mahasiswa : 05 523 360

JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNIK INDUSTRI
UNIVERSITAS ISLAM INDONESIA
YOGYAKARTA
2011

LEMBAR PENGESAHAN PEMBIMBING

ANALISA SISTEM KEAMANAN JARINGAN YANG BERARSITEKTUR MPLS (MULTI PROTOCOL LABEL SWITCHING)

TUGAS AKHIR

Oleh :

Nama : Aan Kurniawan

No. Mahasiswa : 05 523 360

Yogyakarta, 31 Maret 2011

Pembimbing,

R. Teduh Dirgahayu, ST., M.Sc

ii

LEMBAR PENGESAHAN PENGUJI

ANALISA SISTEM KEAMANAN JARINGAN YANG BERARSITEKTUR MPLS (MULTI PROTOCOL LABEL SWITCHING)

TUGAS AKHIR

Oleh :

Nama : Aan Kurniawan

No. Mahasiswa : 05 523 360

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

Yogyakarta, 31 Maret 2011

Tim Penguji,

R. Teduh Dirgahayu, ST., M.Sc.
Ketua

Hendrik, ST., M.Eng.
Anggota I

Ari Sujarwo, S.Kom.
Anggota II

Mengetahui,
Ketua Jurusan Teknik Informatika
Universitas Islam Indonesia

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

iii

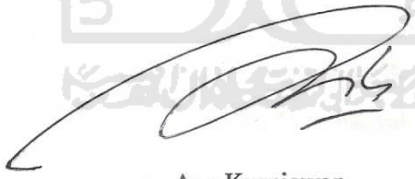
LEMBAR PERNYATAAN KEASLIAN
HASIL TUGAS AKHIR

Saya yang bertandatangan di bawah ini,

Nama : Aan Kurniawan
No. Mahasiswa : 05 523 360
Jurusan : Teknik Informatika

Menyatakan bahwa seluruh komponen dan isi dalam Laporan Tugas Akhir ini adalah hasil karya saya sendiri. Apabila di kemudian hari terbukti bahwa ada beberapa bagian dari karya ini adalah bukan hasil karya saya sendiri, maka saya siap menanggung resiko dan konsekuensi apapun.
Demikian pernyataan ini saya buat, semoga dapat dipergunakan sebagaimana mestinya.

Yogyakarta, 31 Maret 2011



Aan Kurniawan

iv

HALAMAN PERSEMBAHAN

Skripsi ini saya persembahkan untuk,

Allah SWT, yang telah memberikan kehidupan dan semua rahmat karunia ini.

Nabi Muhammad SAW, yang menjadi pedoman disetiap langkah hidupku.

Mama dan Papa yang Aku Sayangi, yang selalu memberikan segalanya, kasih sayang maupun bimbingan baik materi, moril dan spiritual.

Semua saudara saya, yang selalu memberikan semangat dan dukungan tanpa henti.

Saudara Satu Angkatanku Alien'05, yang menjadi keluarga baruku dan bagian hidup yang takkan terlupakan.

Keluarga Besar Informatika UII, yang memberikan banyak pelajaran hidup dan semua kenangan yang indah.

Pak Teduh, yang menjadi pembimbing saya dan memberikan semua pelajaran berharga dalam penyelesaian tugas akhir saya.

Semua Sahabat Saya, yang selalu memberikan semua yang terbaik.

KATA PENGANTAR

Assalamualaikum Wr. Wb.

Segala puji syukur milik Allah SWT, atas berkat Rahmat dan Hidayah-Nya penulis dapat melaksanakan skripsi dengan judul "**Analisa Sistem Keamanan Jaringan yang Berarsitektur MPLS (Multi Protocol Label Switching)**" untuk menyelesaikan studi di Universitas Islam Indonesia sehingga dapat meraih gelar Sarjana Teknik Informatika.

Penulisan skripsi ini dapat terlaksana atas doa, bantuan, dan dorongan dari beberapa pihak, untuk itu penulis sangat mengucapkan terima kasih kepada :

1. Keluargaku tercinta yaitu kedua orang tuaku yang senantiasa memberikan doa, nasihat, dan kasih sayang yang selalu mengiringi hingga selesainya skripsi ini.
2. Bapak R. Teduh Dirgahayu, ST., M.Sc. selaku dosen pembimbing yang telah memberikan pengarahan, bimbingan, serta saran-saran selama penelitian sampai penyusunan skripsi ini.
3. Bapak/Ibu dosen penguji yang telah memberikan kritik dan masukkan atas kesempurnaan naskah skripsi ini.
4. Semua pihak yang telah banyak membantu penulisan skripsi ini yang tidak bisa disebutkan satu per satu.

Penulis hanya dapat mengucapkan terima kasih atas bantuannya dalam penulisan skripsi ini, semoga mendapatkan pahala yang sebesar-besarnya dan semoga amal ibadahnya diterima Allah SWT.

Akhir kata penulis mohon maaf dengan ketulusan hati seandainya dalam penulisan skripsi ini terdapat kekhilafan. Harapan penulis semoga skripsi ini dapat bermanfaat bagi masyarakat pada umumnya serta perkembangan dan kemajuan ilmu pengetahuan pada khususnya, Amin.

Wassalamualaikum Wr. Wb.

Yogyakarta, Januari 2011

Penulis,

Aan Kurniawan

SARI

Perkembangan teknologi jaringan komputer yang terjadi saat ini sangatlah pesat, terutama pada perkembangan teknologi komunikasi data. Hal ini membuat keamanan paket data menjadi suatu yang sangat penting karena komunikasi data tersebut terjadi pada jaringan komputer yang bersifat publik. Perkembangan ini menuntut pada perkembangan sistem keamanan dalam pengiriman paket data yang lebih baik dan mudah untuk diaplikasikan. *Multi Protocol Label Switching* atau yang sering disebut MPLS adalah sebuah teknologi baru yang bertujuan untuk memberikan alternatif lain dalam proses pengiriman paket data pada suatu jaringan komputer. MPLS merupakan gabungan dari kelebihan pengiriman paket data pada lapisan 2 dengan kelebihan-kelebihan *routing* pada lapisan 3 di dalam model lapisan OSI. MPLS melakukan pelabelan pada paket data yang akan di kirim dan akan diteruskan sampai ke tujuan akhir sehingga menawarkan keamanan yang lebih pada paket data tersebut. Penelitian ini bertujuan untuk mengetahui sejauh mana kelebihan dari suatu sistem keamanan jaringan yang berarsitekturkan MPLS, yaitu dengan cara menganalisa paket data yang ada di dalam sistem jaringan MPLS yang akan disimulasikan menggunakan perangkat lunak GNS3 dan Virtualbox.

Penelitian ini dilakukan melalui beberapa tahapan, yang pertama yaitu dengan mensimulasikan suatu jaringan komputer yang berarsitektur MPLS menggunakan alat bantu berupa perangkat lunak GNS3 sebagai *simulator router* Cisco dan Virtualbox sebagai *simulator* PC *client*, setelah simulasi sistem selesai, kemudian dilakukan analisis paket data menggunakan alat bantu Wireshark sebagai *packet sniffer* yang bertugas menangkap dan mencatat paket data yang dilewatkan pada saat PC *client* dan *router* saling berkomunikasi.

Dari hasil penelitian dapat ditarik kesimpulan bahwa suatu sistem jaringan yang berarsitekturkan MPLS hanya melakukan *enkapsulasi* paket datanya saja dan tidak melakukan *enkripsi* pada paket datanya tersebut, namun MPLS memiliki sistem keamanan tersendiri berupa VPN yang berfungsi sebagai sebuah *tunneling* yang menciptakan lorong antar jaringannya sendiri sehingga pertukaran paket data yang terjadi lebih aman karena paket data tidak akan dibocorkan keluar dari VPN yang telah didefinisikan terlebih dahulu.

Kata kunci : keamanan, MPLS, paket data.

TAKARIR

Address	alamat
Dedicated	terdedikasikan
Field	bagian dari sebuah record yang terdiri dari sebuah data yang berisi informasi yang saling berelasi di dalam record tersebut.
Interface	antarmuka pada komputer.
Image	suatu representasi keadaan visual.
Oktet	8 bit pada IP address.
Private	bersifat pribadi.
Router	alat penghubung antara LAN dan Internet yang merutekan transmisi antara keduanya.
Troubleshooting	memecahkan masalah.
Packet sniffer	alat penangkap paket data yang digunakan pada jaringan komputer.
Protocol analysis	alat penganalisa protokol pada jaringan komputer.

DAFTAR ISI

DAFTAR GAMBAR

DAFTAR TABEL

BAB I
PENDAHULUAN

1.1 LATAR BELAKANG

Teknologi jaringan komputer yang berkembang saat ini semakin lama akan semakin maju. Hal ini menuntut perbaikan akan setiap infrastruktur di dalam suatu jaringan untuk menyediakan layanan yang beraneka ragam bentuk dan karakternya, memiliki kapasitas yang tinggi sesuai kebutuhan yang berkembang, mudah diakses kapan saja dan dimana saja, serta beradaptasi dengan peningkatan penggunaan jaringan komputer.

Perkembangan jaringan komputer saat ini memberikan dampak yang besar dalam perkembangan komunikasi data dan peningkatan pengguna jaringan komputer, karena telah memberikan segala kemudahan dalam komunikasi data tanpa batasan ruang dan waktu, sehingga hal ini berdampak ketergantungan bagi pengguna jaringan komputer untuk selalu melakukan komunikasi data melalui media yang menggunakan jaringan komputer contohnya seperti *Internet*. Perkembangan komunikasi data yang pesat membuat sistem keamanan pada komunikasi data menjadi sesuatu yang sangat penting untuk diperhatikan, karena semua komunikasi data yang dilakukan terjadi pada jaringan yang bersifat publik dan dapat di akses oleh semua orang yang terhubung pada jaringan komputer tersebut.

IP sebagai teknologi terdahulu dirasa memiliki banyak kekurangan, baik itu dalam hal pengiriman paket data ataupun manajemen *bandwith*, serta dalam pengamanan paket datanya teknologi IP masih memerlukan penambahan pengamanan khusus contohnya seperti IPsec, namun teknologi IP memiliki kelebihan dari segi skalabilitas yang membuat teknologi ini lebih murah. MPLS memperbaiki kinerja pengiriman suatu paket data dengan cara

melakukan pelabelan pada setiap paket data yang akan dikirim dan memberikan prioritas paket yang harus sampai tujuan terlebih dahulu. Hal ini menyebabkan tingkat keamanan pada jaringan MPLS lebih baik, karena paket hanya akan sampai pada tujuan akhir sesuai dengan label yang telah diberikan pada masing–masing paket data.

Penggunaan jaringan yang sama untuk kebutuhan yang berbeda-beda sering menimbulkan permasalahan diantaranya penumpukan trafik pada jaringan, untuk itu penerapan Multi-Protocol Label Switching (MPLS) menjadikan jaringan bersifat *private* dan lebih aman karena MPLS menggunakan VPN sebagai dasar arsitektur jaringannya.

1.2 RUMUSAN MASALAH

Rumusan masalah yang diangkat pada penelitian tugas akhir ini adalah bagaimana menganalisis tingkat keamanan paket data pada suatu sistem jaringan komputer yang berasitektur MPLS.

1.3 BATASAN MASALAH

Penelitian ini dibatasi pada beberapa masalah:

1. Analisis dilakukan pada simulator GNS3 yang terhubung dengan Virtualbox
2. Analisis dilakukan pada aktivitas data pada arsitektur MPLS
3. Dilakukan pengujian keamanan hanya untuk melihat sejauh mana ancaman keamanan yang mungkin terjadi.

1.4 TUJUAN PENELITIAN

Tujuan yang ingin dicapai dalam penelitian ini yaitu :

1. Penerapan simulasi jaringan yang berasitektur MPLS-VPN menggunakan virtualbox dan GNS3

2. Mengintegrasikan MPLS pada virtualisasi VPN.
3. Meneliti sistem keamanan pada jaringan MPLS-VPN.

1.5 MANFAAT PENELITIAN

Penelitian ini ditulis dengan harapan dapat memberikan manfaat yang luas, khususnya untuk penulis sendiri maupun untuk dunia akademik secara umum. Adapun beberpa manfaat yang diharapkan adalah sebagai berikut :

1. Manfaat untuk penulis :
   a. Mampu membangun suatu jaringan MPLS-VPN.
   b. Mengerti perbedaan antara VPN pada jaringan MPLS dengan VPN yang bukan di bangun dengan MPLS.
   c. Mampu menganalisis paket data suatu sistem jaringan komputer.
2. Manfaat akademik :
   a. Menjadi referensi belajar untuk meningkatkan pengetahuan khususnya di bidang jaringan komputer.
   b. Menjadi bahan dasar penelitian untuk perkembangan teknologi dibidang jaringan komputer.

1.6 HIPOTESIS

Hipotesis pada penelitian ini, yaitu sistem jaringan komputer berarsitektur MPLS dapat memperketat sistem keamanan jaringan komputer.

1.7 SISTEMATIKA PENULISAN

Sistematika penulisan laporan tugas akhir ini adalah sebagai berikut :

BAB I PENDAHULUAN

Berisi latar belakang, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, hipotesis, dan sistematika penulisan.

BAB II LANDASAN TEORI

Berisi dengan tinjauan pustaka dan teori dasar penelitian.

BAB III METODOLOGI

Berisikan uraian tentang langkah–langkah penelitian, analisis kebutuhan perangkat keras dan perangkat lunak, perancangan sistem, analisis sistem, serta rencana pengujian.

BAB IV HASIL DAN PEMBAHASAN

Berisikan uraian tentang hasil yang dicapai, bagaimana hasil dapat dicapai dan pembahasan mengapa hasil tersebut dapat dicapai serta pengujian terhadap sistem yang telah dibuat.

BAB V SIMPULAN DAN SARAN

Memuat simpulan–simpulan dari hasil analisis pada bagian sebelumnya dan berisi saran-saran yang perlu diperhatikan berdasarkan keterbatasan-keterbatasan yang ditemukan dan asumsi-asumsi yang dibuat selama penelitian.

UNIVERSITAS ISLAM INDONESIA

BAB II
LANDASAN TEORI

2.1. Jaringan Komputer

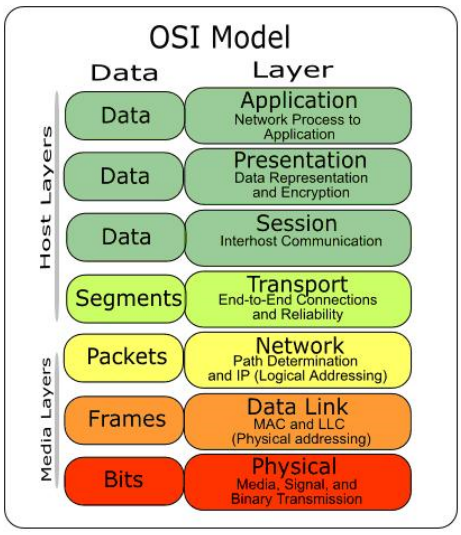
2.1.1. Pengertian Jaringan Komputer

Jaringan komputer adalah sebuah interkoneksi (saling keterhubungan) antara kelompok-kelompok komputer dengan komputer lain.

Dengan jaringan komputer, komputer-komputer akan menjadi satu kesatuan sehingga para penggunanya bisa saling mengakses dan bertukar data tanpa harus berpindah membawa sebuah disket atau media USB dari komputer satu ke komputer lainnya. Selain itu, jaringan komputer juga dapat disambungkan ke Internet (Komputer, 2009).

2.1.2. MODEL OSI LAYER

Model *Open Systems Interconnection* (OSI) diciptakan oleh *International Organization Of Standarization* yang menyediakan kerangka logika terstruktur, bagaimana proses komunikasi data berinteraksi melalui jaringan. Secara lengkap gambaran Model OSI dapat di lihat pada gambar 1.1.



Gambar 1.1 Model OSI

Standart ini dikembangkan untuk industri komputer agar komputer dapat berkomunikasi pada jaringan berbeda secara efisien. Berikut adalah lapisan OSI beserta fungsi-fungsinya yang dapat di lihat pada tabel 1.17 (Irianto, 2011) :

Tabel 1.1 7 Lapisan Model OSI

7	Application	Menyediakan jasa untuk aplikasi pengguna. Lapisan ini bertanggungjawab atas pertukaran informasi antara program komputer,seperti program e-mail, dan service lain yang jalan di jaringan, seperti server printer atau aplikasi komputer lainnya
6	Presentation	Bertanggung jawab bagaimana data dikonversi dan diformat untuk transfer data.Contoh konversi format text ASCII untuk dokumen,.gif dan JPG untuk gambar. Lapisan ini membentuk kode konversi, translasi data, enkripsi dan konversi.
5	Session	Menentukan bagaimana dua terminal menjaga, memelihara dan mengatur koneksi serta bagaimana mereka saling berhubungan satu sama lain.
4	Transport	Bertanggung jawab membagi data menjadi segmen, menjaga koneksi logika end-to-end antar terminal, dan menyediakan penanganan kesalahan (error handling).
3	Network	Bertanggung jawab menentukan alamat jaringan, menentukan rute yang harus diambil selama perjalanan, dan menjaga antrian trafik di jaringan. Data pada lapisan ini berbentuk paket.
2	Data link	Menyediakan link untuk data, memaketkannya menjadi frame yang berhubungan dengan perangkat keras kemudian diangkut melalui media. komunikasinya dengan kartu jaringan, mengatur komunikasi lapisan physical antara sistem koneksi dan penanganan kesalahan.
1	Physical	Bertanggung jawab atas proses data menjadi bit dan mentransfernya melalui media, seperti kabel, dan menjaga koneksi fisik antar sistem.

2.1.3 TCP/IP

Transmision Control Protocol and Internet Protocol atau disingkat TCP/IP yang merupakan sekelompok protokol yang mengatur komunikasi data di Internet. Komputer-komputer yang terhubung ke Internet saling berkomunikasi dengan protokol ini.

Karena menggunakan protokol komunikasi yang sama, yaitu protokol TCP/IP, perbedaan jenis komputer dan sistem operasi tidak menjadi masalah. Jika

sebuah komputer menggunakan protokol TCP/IP dan terhubung ke Internet, maka komputer tersebut dapat berkomunikasi dengan komputer manapun yang terhubung ke Internet. Protokol TCP/IP, meliputi beberapa protokol lain, seperti TCP (*Transmission Control Protocol*), UDP (*User Datagram Protocol*), IP (*Internet Protocol*) dan ICMP (*Internet Control Message Protocol*). Berikut adalah beberapa elemen konfigurasi umum TCP/IP dan tujuannya:

- IP *address*

  Merupakan sebuah string unik yang dituliskan dalam angka desimal yang dibagi dalam empat segmen. Tiap-tiap segmen tersebut merepresentasikan 8 bit dari alamat yang memiliki panjang 32 bit untuk keseluruhannya.

- Netmask

  Merupakan singkatan dari *Subnet Mask* adalah angka *biner* 32 bit yang digunakan untuk membedakan network ID dengan host ID, apakah berada di jaringan lokal atau luar. Berikut adalah *subnet mask* standar dari tiap kelas IP *address* :

  a. Kelas A = 255.0.0.0
  b. Kelas B = 255.255.0.0
  c. Kelas C = 255.255.255.0

- Network Address

  Network address merepresentasikan porsi jaringan dari alamat IP. Misalnya, host 12.126.1.2 di jaringan kelas A memiliki *network address* 12.0.0.0.

- Broadcast Address

  Broadcast Address merupakan alamat IP yang memungkinkan data jaringan dikirimkan ke semua host di dalam sebuah jaringan. Alamat broadcast biasanya diset untuk jaringan tertentu saja yang telah dipisahkan dengan subnetting pada subnet masknya, misal alamat IP 192.168.1.0 akan memiliki alamat broadcast 192.168.1.255.

- Gateway Address

  Merupakan alamat IP yang harus dilewati oleh semua komputer di suatu jaringan jika ingin berkomunikasi dengan host dijaringan lain. Contohnya alamat IP 192.168.1.5 memiliki alamat gateway 192.168.1.1.

- Nameserver Address

  Nameserver *address* menunjukan IP address dari Domain Name Service (DNS) yang bertujuan menerjemahkan nama host ke alamat IP (Komputer, 2009).

2.1.4 IP ADDRESS

IP *address* atau alamat IP adalah pengalamatan yang digunakan untuk mengidentifikasi *interface* jaringan pada suatu komputer. IP *address* terdiri atas dua versi, yaitu IPv4 dan IPv6.

Alamat IP versi 4 berupa sekelompok bilangan biner 32 bit yang dibagi menjadi 4 bagian dan masing-masing bagian terdiri dari 8 bit. Untuk memudahkan pembacaan suatu alamat IP, maka penamaan yang digunakan adalah berdasarkan bilangan desimal.

Sedangkan alamat IP versi 6 berupa sekelompok bilangan hexadesimal sepanjang 128 bit yang dipisahkan oleh tanda dua setiap 8 bit.

Alamat IP versi 4 yang banyak digunakan saat ini terdiri dari tiga kelas. Kelas-kelas ini ditentukan berdasarkan jumlah *oktet* awal yang digunakan sebagai identitas jaringan. Tiga kelas tersebut adalah :

- **Kelas A**, merupakan kelas yang memiliki kapasitas jumlah alamat IP terbanyak adalah IP *address* yang bagian awalnya berada di antara angka 1 hingga 126.

  Format : 0xxxxxxx.yyyyyyyy.yyyyyyyy.yyyyyyyy
  Subnetmask default : 255.0.0.0
  Kisaran : 0.0.0.0 – 127.255.255.255
  Jumlah host : 16.777.214

- **Kelas B**, diidentifikasi dengan 2 *oktet*, di mana jangkauan *oktet* pertamanya antara 128 hingga 192.

  Format : 10xxxxxx.xxxxxxxx.yyyyyyyy.yyyyyyyy
  Subnetmask default : 255.255.0.0
  Kisaran : 128.0.0.0 – 191.255.255.255
  Jumlah Host : 65.532

- **Kelas C**, merupakan kelas IP yang memiliki kapasitas jumlah alamat IP paling sedikit diantara kedua kelas diatas yang diidentifikasi dengan 3 *oktet*, di mana jangkauan *oktet* pertamanya antara 192 hingga 223.

  Format : 110xxxxx.xxxxxxxx.xxxxxxxx.yyyyyyyy
  Subnetmask default : 255.255.255.0
  Kisaran : 192.0.0.0 – 232.255.255.255
  Jumlah host : 254
  x = network ID
  y = host

Tiap-tiap IP *address* hanya bisa berhubungan jika mereka berada di satu jaringan (Komputer, 2009).

2.1.5 SUBNETTING

Konsep *subnetting* muncul akibat kekhawatiran menipisnya jumlah alamat IP yang ada di Internet. *Subnetting* adalah pembagian logika sebuah jaringan besar, dalam hal ini memecah subnet mask menjadi bagian-bagian lagi sesuai kebutuhan sebuah jaringan dalam menggunakan host didalam jaringannya.

Subnet mask terdiri dari bilangan-bilangan biner, misalnya 255.0.0.0. Jika dikonversi ke biner, maka menjadi 11111111.00000000.00000000.00000000. Penulisannya bisa dinyatakan sebagai 255.0.0.0/8. Angka 8 menunjukan jumlah bit aktif (angka1). Contoh lain adalah 255.255.255.224/27. Jika *subnet mask* ini dikonversi ke biner maka akan menghasilkan 11111111.11111111.11111111.11100000. Angka 27 menunjukan jumlah bit aktif (angka 1) mulai dari *oktet* yang pertama hingga keempat (Komputer, 2009).

2.2 MPLS

Multiprotocol Label Switching (MPLS) adalah teknologi penyampaian paket data yang berbasis paket. MPLS beroperasi pada model lapisan OSI yang secara umum berada diantara definisi tradisional dari lapisan 2 (data link) dan lapisan 3 (network), dan sering disebut sebagai lapisan 2,5.

Teknologi MPLS diterapkan dengan tujuan untuk meningkatkan kemampuan dari teknologi jaringan IP. Ide dasar dari pengembangan teknologi MPLS adalah menggunakan label untuk melakukan mekanisme *switching* ditingkat IP. Hal ini berbeda dengan teknologi IP yang menggunakan pengalamatan IP sebagai dasar mekanisme *switching*.

Di dalam jaringan yang menggunakan protokol MPLS, paket yang masuk ke dalam jaringan MPLS terlebih dahulu diberi label. Berdasarkan label yang diberikan ini maka jaringan yang menggunakan protokol MPLS akan memperlakukan paket tersebut sesuai dengan nilai yang melekat pada label tersebut paket mana yang harus sampai lebih dahulu dan paket mana yang harus sampai setelahnya (Ilyas, 2011).

2.3 MPLS-VPN

Salah satu kemampuan MPLS adalah membentuk *tunnel* yang melintasi networknya. Kemampuan ini membuat MPLS dapat berfungsi sebagai *platform* alami untuk membangun *virtual private network* (VPN).

VPN yang dibangun dengan menggunakan MPLS berbeda dengan VPN yang dibangun berdasarkan dengan teknologi IP, karena dengan MPLS, VPN dibangun membentuk isolasi trafik yang terpisah dan tidak dapat dibocorkan ke luar lingkup VPN yang didefinisikan.

VPN pada MPLS menggunakan *virtual routing forwarding* (vrf) untuk memisahkan *routing target* pada masing-masing *tunnel* yang dibuat, sehingga terbentuk sebuah lorong khusus yang bersifat *private* untuk saling terhubung antara *routing target*-nya saja walaupun berada pada suatu jaringan yang bersifat publik (Wastuwibowo, 2003).

2.4 VIRTUALBOX

VirtualBox adalah sebuah aplikasi virtual mesin yang digunakan untuk menginstall Sistem Operasi (SO) lain, dan dijalankan bersamaan di atas sistem operasi induknya. VirtualBox adalah aplikasi open source keluaran Sun MicroSystem yang ditargetkan untuk Server dan pengguna komputer desktop.

Saat ini virtualbox mampu berjalan pada sistem operasi Windows, Linux, Macintosh dan OpenSolaris (Coorporation, 2004).

2.5 GNS3

GNS3 merupakan sebuah program *graphical network simulator* yang dapat mensimulasikan topologi jaringan kompleks. GNS3 dapat berjalan diatas sistem operasi seperti Linux dan Windows. Prinsip kerja dari GNS3 adalah mengemulasi Cisco IOS pada komputer, sehingga PC dapat berfungsi layaknya beberapa *switch* atau *router* dengan cara mengaktifkan fungsi dari *Ethernet Switch Card* (Saputro, 2010).

2.6 WIRESHARK

Wireshark merupakan salah satu perangkat analisa jaringan yang sering disebut juga *protocol analysis tool* atau *packet sniffer*. Wireshark dapat digunakan untuk troubleshooting, analisis, pengembangan perangkat lunak dan protokol, serta untuk keperluan pendidikan. Wireshark merupakan perangkat lunak gratis yang sebelumnya dikenal dengan nama Ethereal

Packet sniffer sendiri dapat diartikan sebagai sebuah perangkat lunak atau alat yang memiliki kemampuan untuk menghadang dan melakukan catatan terhadap *traffic* data dalam jaringan.

Wireshark sebagai salah satu *packet sniffer* diprogram sedemikian rupa untuk mengenali berbagai macam protokol jaringan. Wireshark mampu menampilkan hasil dari enkapsulasi dan *field* yang ada dalam *Protocol Data Unit* (Netlab, 2010).

ISLAM
UNIVERSITAS
INDONESIA

BAB III
METODOLOGI

3.1. Kebutuhan Perangkat

3.1.1 Analisa Kebutuhan Perangkat Lunak

a) **Windows Vista Home Basic**

b) **Virtualbox**

Virtualbox yang digunakan adalah Virtualbox ver 3.1.4.

c) **GNS3**

GNS3 yang digunakan adalah GNS3 ver 0.7.3. Dalam penelitian ini *router* yang digunakan adalah seri c3600 dengan menggunakan Cisco IOS Image c3640-jk9s-mz.124-16.

d) **Wireshark**

Wireshark yang digunakan adalah Wireshark ver 1.2.8

3.1.2 Perangkat Keras yang dibutuhkan

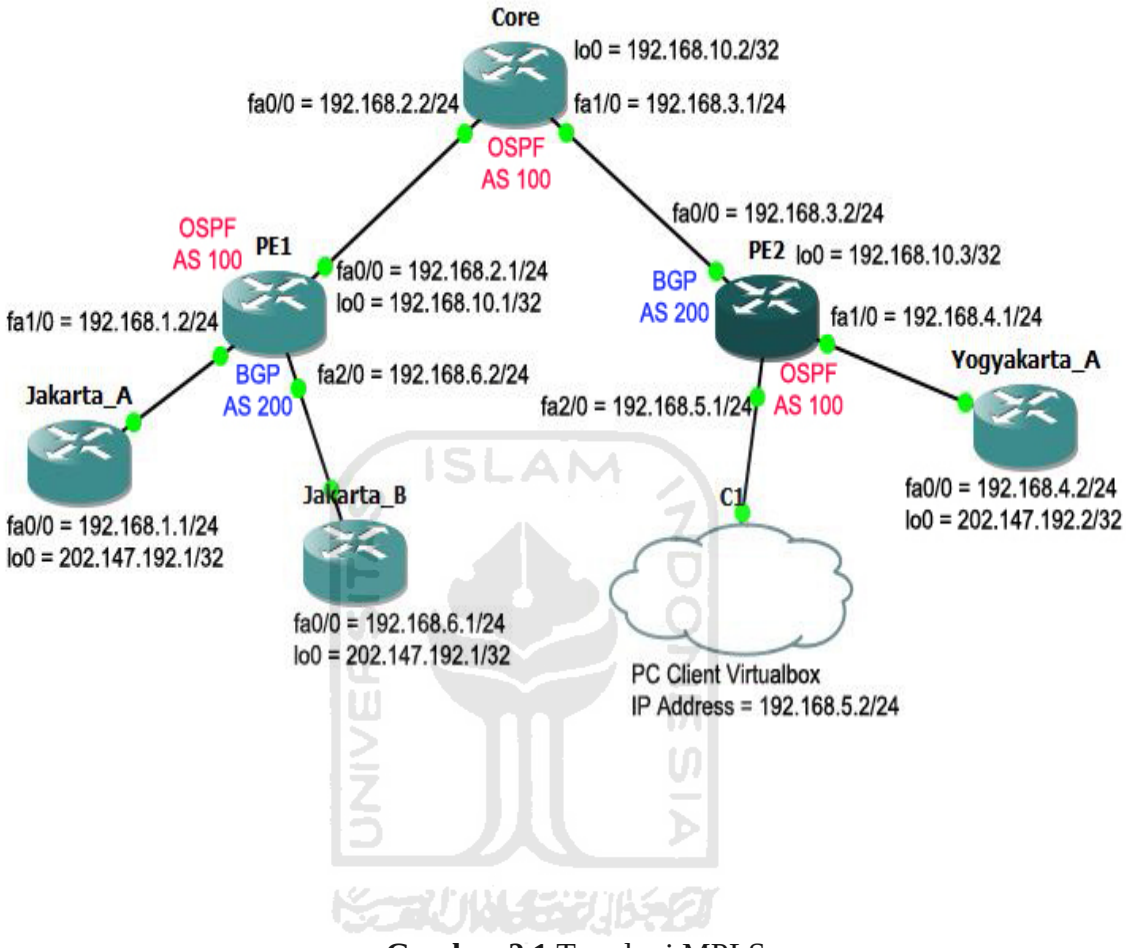
Kebutuhan perangkat keras yang digunakan dalam mengerjakan penelitian ini adalah sebagai berikut :

- Laptop Dell Inspiron 1420
- Intel Core 2 duo T5750 @2.00 GHz
- Hardisk 120 Gb
- 2 Gb RAM

3.2 PERANCANGAN SISTEM

Penelitian ini akan melakukan simulasi sebuah perusahaan yang memiliki cabang di daerah dan membutuhkan jaringan komputer untuk saling berhubungan antara kantor pusat dan kantor cabangnya. Berikut gambaran topologi sederhana yang akan disimulasikan melalui GNS3 yang dapat dilihat pada gambar 2.1. Router Core, PE1 dan PE2 berada pada jaringan publik sedangkan *router* Jakarta dan Yogyakarta merupakan *router* CE, cloud dalam

topologi ini digunakan untuk menghubungkan GNS3 dengan pc *client* dari Virtualbox.



Gambar 2.1 Topologi MPLS

Dalam arsitektur MPLS terdiri dari 3 jenis *router* yang berperan penting, yaitu :

- Router Core
  - ~ merupakan *router* yang berada pada jaringan publik yang tidakberhubungan langsung dengan *router* CE, *router* ini bertanggung jawab untuk fungsi *routing* dan *forwarding*.

- Router PE
  - ~ *Provider Edge* (PE) merupakan *router* yang berada pada jaringan publik yang terletak diantara *router* Core dan *router* CE, *router* ini bertugas sebagai penghubung kedua *router* tersebut.

- Router CE
  - ~ *Customer Edge* (CE) merupakan *router* yang terhubung langsung ke *client*.

Berikut adalah tabel konfigurasi interface dari semua *router* pada topologi ini yang secara lengkap dapat dilihat pada tabel 2.1

Tabel 2.1 Daftar tabel routing dari semua router yang ada di topologi

Interface / Router	lo0	Fa0/0	Fa1/0	Fa2/0
Core	192.168.10.2/32	192.168.2.2/24	192.168.3.1/24	-
PE1	192.168.10.1/32	192.168.2.1/24	192.168.1.2/24	192.168.6.2/24
PE2	192.168.10.3/32	192.168.3.2/24	192.168.4.1/24	192.168.5.1/24
Jakarta_A	202.147.192.1/32	192.168.1.1/24	-	-
Jakarta_B	202.147.192.1/32	192.168.6.1/24	-	-
Yogyakarta_A	202.147.192.2/32	192.168.4.2/24	-	-

3.2.1. KONFIGURASI

3.2.1.1 Konfigurasi Interface

Langkah pertama yang harus dilakukan adalah melakukan konfigurasi semua interface di ketiga *router* inti yaitu, *router* Core, *router* PE1, dan *router* PE2. Berikut adalah konfigurasi untuk interface lo0,fa0/0, dan fa1/0 pada *router* Core, konfigurasi ini dilakukan juga untuk kedua *router* PE:

Router Core :

```
Core>en
Core#conf t
Core(config)#int lo0
Core(config-if)#ip add 192.168.10.2 255.255.255.255
```

```
Core(config-if)#no sh
Core(config-if)#end
Core#conf t
Core(config)#int fa0/0
Core(config-if)#ip add 192.168.2.2 255.255.255.0
Core(config-if)#no sh
Core(config-if)#end
Core#conf t
Core(config)#int fa1/0
Core(config-if)#ip add 192.168.3.1 255.255.255.0
Core(config-if)#no sh
Core(config-if)#end
Core#wr
```

Interface fa1/0 dan 2/0 pada *router* PE1 dan PE2 dikosongkan untuk ip vrf virtual. Virtual Routing Forwarding (VRF) merupakan suatu virtual *router* yang digunakan untuk memisahkan dua VPN berbeda yang ada pada sistem jaringan ini.

3.2.1.2 Konfigurasi Routing OSPF

Langkah berikutnya adalah melakukan konfigurasi routing OSPF. *Open Shotest Path First* (OSPF) merupakan routing dinamik yang digunakan oleh MPLS untuk menyebar informasi *bandwith*, mendistribusikan label, dan menghitung jalur atau path dalam jaringan. Berikut adalah konfigurasi OSPF pada *router* PE1, konfigurasi ini dilakukan juga pada *router* PE2 dan Core :

Router PE1 :

```
PE1>en
PE1#conf t
PE1(config)#router ospf 100
PE1(config-router)#network 192.168.2.0 0.0.0.255 area 0
PE1(config-router)#network 192.168.10.1 0.0.0.0 area 0
PE1(config-router)#end
```

```
PE1#wr
```

Pengecekan dynamic routing dilakukan dengan menggunakan perintah berikut :

#sh ip route

Pengecekan ini dilakukan di semua *router*, jika semua alamat IP sudah terpopulasi artinya terdapat pertukaran informasi mengenai routing OSPF pada *router* PE dan Core, maka dapat dilakukan ping antara alamat IP interface loopback pada semua *router*.

3.2.1.3 Konfigurasi Routing BGP

Langkah berikutnya adalah melakukan konfigurasi routing BGP. *Border Gateway Protocol* (BGP) digunakan oleh MPLS untuk mendistribusikan informasi tentang VPN hanya ke *router* dalam VPN yang sama, sehingga terjadi pemisahan trafik. Berikut adalah konfigurasi BGP pada *router* PE1 dan PE2.

Router PE1 :

```
PE1>en
PE1#conf t
PE1(config)#router BGP 200
PE1(config-router)#no synchronization
PE1(config-router)#neighbor 192.168.10.3 remote-as 200
PE1(config-router)#neighbor 192.168.10.3 update-source
loopback0
PE1(config-router)#no auto-summary
PE1(config-router)#end
PE1#wr
```

Router PE2 :

```
PE2>en
PE2#conf t
PE2(config)#router BGP 200
PE2(config-router)#no synchronization
PE2(config-router)#neighbor 192.168.10.1 remote-as 200
```

```
PE2(config-router)#neighbor 192.168.10.1 update-source
loopback0
PE2(config-router)#no auto-summary
PE2(config-router)#end
PE2#wr
```

3.2.1.4 Konfigurasi MPLS

Langkah berikutnya adalah mengaktifkan MPLS.

Router PE1 :

```
PE1>en
PE1#conf t
PE1(config)#ip cef
PE1(config)#mpls label protocol ldp
PE1(config)#mpls ldp router-id lo0 force
PE1(config)#int fa0/0
PE1(config-if)#ip add 192.168.2.1 255.255.255.0
PE1(config-if)#mpls ip
PE1(config-if)#end
PE1#wr
```

Router PE2 :

```
PE2>en
PE2#conf t
PE2(config)#ip cef
PE2(config)#mpls label protocol ldp
PE2(config)#mpls ldp router-id lo0 force
PE2(config)#int fa0/0
PE2(config-if)#ip add 192.168.3.2 255.255.255.0
PE2(config-if)#mpls ip
PE2(config-if)#end
PE2#wr
```

Router Core :

```
Core>en
Core#conf t
Core(config)#ip cef
Core(config)#mpls label protocol ldp
Core(config)#mpls ldp router-id lo0 force
Core(config)#int fa0/0
Core(config-if)#ip add 192.168.2.2 255.255.255.0
Core(config-if)#mpls ip
Core(config-if)#end
Core#conf t
Core(config)#ip cef
Core(config)#mpls label protocol ldp
Core(config)#mpls ldp router-id lo0 force
Core(config)#int fa1/0
Core(config-if)#ip add 192.168.3.1 255.255.255.0
Core(config-if)#mpls ip
Core(config-if)#end
Core#wr
```

Melakukan pengecekan MPLS di setiap *router* dengan menjalankan perintah sebagai berikut:

#sh mpls ldp neighbor

Jika konfigurasi berhasil maka akan terdapat informasi tentang pertukaran routing MPLS table pada masing-masing *router*.

3.2.1.5 Konfigurasi Router Virtual

Langkah Berikutnya adalah melakukan konfigurasi *router* virtual.

Router PE1 :

```
PE1>en
PE1#conf terminal
PE1(config)#ip vrf vpn1
```

```
PE1(config-vrf)#rd 100:1
PE1(config-vrf)#route-target export 100:1
PE1(config-vrf)#route-target import 100:1
PE1(config-vrf)#end
PE1#conf t
PE1(config)#ip vrf vpn2
PE1(config-vrf)#rd 100:2
PE1(config-vrf)#route-target export 100:2
PE1(config-vrf)#route-target import 100:2
PE1(config-vrf)#end
PE1#conf t
PE1(config)#int fa1/0
PE1(config-if)#ip vrf forwarding vpn1
PE1(config-if)#ip add 192.168.1.2 255.255.255.0
PE1(config-if)#end
PE1#conf t
PE1(config)#int fa2/0
PE1(config-if)#ip forwarding vpn2
PE1(config-if)#ip add 192.168.6.2 255.255.255.0
PE1(config-if)#end
PE1#wr
```

Router PE2 :

```
PE2#conf t
PE2(config)#ip vrf vpn1
PE2(config-vrf)#rd 100:1
PE2(config-vrf)#route-target export 100:1
PE2(config-vrf)#route-target import 100:1
PE2(config-vrf)#end
PE2#conf t
PE2(config)#ip vrf vpn2
```

```
PE2(config-vrf)#rd 100:2
PE2(config-vrf)#route-target export 100:2
PE2(config-vrf)#route-target import 100:2
PE2(config-vrf)#end
PE2#conf t
PE2(config)#int fa1/0
PE2(config-if)#ip vrf forwarding vpn1
PE2(config-if)#ip add 192.168.4.1 255.255.255.0
PE2(config-if)#end
PE2#conf t
PE2(config)#int fa2/0
PE2(config-if)#ip forwarding vpn2
PE2(config-if)#ip add 192.168.5.1 255.255.255.0
PE2(config-if)#end
PE2#wr
```

Untuk melakukan pengecekan konfigurasi ini, dijalankan perintah berikut pada salah satu *router* PE:

#sh ip vrf vpn1

Jika terdapat interface connected, maka konfigurasi ini berhasil

3.2.1.6 Konfigurasi Static Routing di router virtual

Langkah berikutnya adalah melakukan konfigurasi static routing di *router* virtual di tiap *router* PE untuk interface yang terhubung langsung dengan interface virtual pada *router* PE tersebut. Perintah yang diberikan adalah sebagai berikut.

Router PE1 :

```
PE1>en
PE1#conf t
PE1(config)#ip    route    vrf    vpn1    202.147.192.1
255.255.255.255 192.168.1.1
PE1(config)#ip    route    vrf    vpn2    202.147.192.1
255.255.255.255 192.168.6.1
```

```
PE1(config)#end
PE1#wr
```

Router PE2 :

```
PE2>en
PE2#conf t
PE2(config)#ip    route    vrf    vpn1    202.147.192.2
255.255.255.255 192.168.4.2
PE2(config)#ip    route    vrf    vpn2    202.147.192.2
255.255.255.255 192.168.5.2
PE2(config)#end
PE2#wr
```

Setelah penambahan routing static selesai, dapat dilakukan pengecekan pada salah satu *router* PE dengan menjalankan perintah sebagai berikut :

#sh ip ro vrf vpn1

#sh ip ro vrf vpn2

Jika semua ip yang telah tambahkan tadi ada, berarti konfigurasi tahap ini berhasil,

3.2.1.7 Konfigurasi Routing MP-BGP

Langkah berikutnya adalah melakukan konfigurasi MP-BGP sebagai tunneling antara *router* PE1 dan PE2, dengan menggunakan perintah berikut.

Router PE1 :

```
PE1>en
PE1#conf terminal
PE1(config)#router bgp 200
PE1(config-router)#address-family vpnv4
PE1(config-router)#neighbor 192.168.10.3 activate
PE1(config-router)#neighbor 192.168.10.3 send-community
both
PE1(config-router)#exit-address-family
```

```
PE1(config-router)#end
PE1#wr
```

Router PE2 :

```
PE2>en
PE2#conf t
PE2(config)#router bgp 200
PE2(config-router)#address-family vpnv4
PE2(config-router)#neighbor 192.168.10.1 activate
PE2(config-router)#neighbor 192.168.10.1 send-community
both
PE2(config-router)#exit-address-family
PE2(config-router)#end
PE2#wr
```

Pengecekan di kedua sisi *router* PE dilakukan dengan menjalankan perintah "sh running", konfigurasi berhasil dilakukan apabila terdapat informasi berikut

```
address-family ipv4 vrf vpn2
no synchronization
exit-address-family
!
address-family ipv4 vrf vpn1
no synchronization
exit-address-family
```

Selanjutnya dilakukan penambahan informasi routing static pada setiap *router* PE dengan menjalankan perintah berikut

Router PE1 :

```
PE1>en
PE1#conf t
PE1(config)#router bgp 200
PE1(config-router)#address-family ipv4 vrf vpn1
PE1(config-router-af)#redistribute connected
```

```
PE1(config-router-af)#redistribute static
PE1(config-router-af)#end
PE1#conf t
PE1(config)#router bgp 200
PE1(config-router)#address-family ipv4 vrf vpn2
PE1(config-router-af)#redistribute connected
PE1(config-router-af)#redistribute static
PE1(config-router-af)#end
PE1#wr
```

Router PE2 :

```
PE2>en
PE2#conf t
PE2(config)#router bgp 200
PE1(config-router)#address-family ipv4 vrf vpn1
PE1(config-router-af)#redistribute connected
PE1(config-router-af)#redistribute static
PE1(config-router-af)#end
PE1#conf t
PE1(config)#router bgp 200
PE1(config-router)#address-family ipv4 vrf vpn2
PE1(config-router-af)#redistribute connected
PE1(config-router-af)#redistribute static
PE1(config-router-af)#end
PE1#wr
```

Jika konfigurasi dilakukan dengan tepat, perintah “sh run” akan memberikan informasi tentang *router* BGP secara lengkap, yaitu :

Router PE1 :

```
router bgp 200
no synchronization
bgp log-neighbor-changes
```

```
neighbor 192.168.10.3 remote-as 200
neighbor 192.168.10.3 update-source Loopback0
no auto-summary
!
address-family vpnv4
21
neighbor 192.168.10.3 activate
neighbor 192.168.10.3 send-community both
exit-address-family
!
address-family ipv4 vrf vpn2
redistribute connected
redistribute static
no synchronization
exit-address-family
!
address-family ipv4 vrf vpn1
redistribute connected
redistribute static
no synchronization
exit-address-family
```

Router PE2 :

```
router bgp 200
no synchronization
bgp log-neighbor-changes
neighbor 192.168.10.1 remote-as 200
neighbor 192.168.10.1 update-source Loopback0
no auto-summary
!
address-family vpnv4
```

```
neighbor 192.168.10.1 activate
neighbor 192.168.10.1 send-community both
exit-address-family
!
address-family ipv4 vrf vpn2
redistribute connected
redistribute static
no synchronization
exit-address-family
!
address-family ipv4 vrf vpn1
redistribute connected
redistribute static
no synchronization
exit-address-family
!
```

3.2.1.8 Konfigurasi Router CE

Konfigurasi terakhir adalah penentuan alamat IP pada *router* Jakarta dan Yogyakarta sebagai *router* CE agar dapat terhubung ke jaringan MPLS yang telah dibuat.

Router Jakarta_A :

```
Jakarta_A>en
Jakarta_A#conf t
Jakarta_A(config)#memory-size iomem 5
Jakarta_A(config)#int lo0
Jakarta_A(config-if)#ip add 202.147.192.1
255.255.255.255
Jakarta_A(config-if)#no sh
Jakarta_A(config-if)#end
Jakarta_A#conf t
```

```
Jakarta_A(config)#int fa0/0
Jakarta_A(config-if)#ip add 192.168.1.1 255.255.255.0
Jakarta_A(config-if)#no sh
Jakarta_A(config-if)#end
Jakarta_A#conf t
Jakarta_A(config)#no ip http server
Jakarta_A(config)#ip route 202.147.192.2
255.255.255.255 192.168.1.2
Jakarta_A(config)#ip  route  192.168.4.0  255.255.255.0
192.168.1.2
Jakarta_A(config)#end
Jakarta_A#wr
```

Router Jakarta_B :

```
Jakarta_B>en
Jakarta_B#conf t
Jakarta_B(config)#memory-size iomem 15
Jakarta_B(config)#ip subnet-zero
Jakarta_B(config)#int lo0
Jakarta_B(config-if)#ip add 202.147.192.1
255.255.255.255
Jakarta_B(config-if)#no sh
Jakarta_B(config-if)#end
Jakarta_B#conf t
Jakarta_B(config)#int fa0/0
Jakarta_B(config-if)#ip add 192.168.6.1 255.255.255.0
Jakarta_B(config-if)#no sh
Jakarta_B(config-if)#end
Jakarta_B#conf t
Jakarta_B(config)#ip classless
```

```
Jakarta_B(config)#ip route 202.147.192.2
255.255.255.255 192.168.6.2
Jakarta_B(config)#ip route 192.168.5.0 255.255.255.0
192.168.6.2
Jakarta_B(config)#no ip http server
Jakarta_B(config)#ip pim bidir-enable
Jakarta_B(config)#end
Jakarta_B#wr
```

Router Yogyakarta_A :

```
Yogyakarta_A>en
Yogyakarta_A#conf t
Yogyakarta_A(config)#memory-size iomem 5
Yogyakarta_A(config)#int lo0
Yogyakarta_A(config-if)#ip add 202.147.192.2
255.255.255.255
Yogyakarta_A(config-if)#no sh
Yogyakarta_A(config-if)#end
Yogyakarta_A#conf t
Yogyakarta_A(config)#int fa0/0
Yogyakarta_A(config-if)#ip add 192.168.4.2
255.255.255.0
Yogyakarta_A(config-if)#no sh
Yogyakarta_A(config-if)#end
Yogyakarta_A#conf t
Yogyakarta_A(config)#ip http server
Yogyakarta_A(config)#ip route 0.0.0.0 0.0.0.0
192.168.4.1
Yogyakarta_A(config)#end
Yogyakarta_A#wr
```

PC :

```
set IP static 192.168.5.2
subnet 255.255.255.0
gateway 192.168.5.1
```

Pengecekan hasil konfigurasi dilakukan dengan menggunakan perintah berikut :

```
Jakarta_A#ping 202.147.192.2
```

mengirimkan ping menuju interface loopback0 Yogyakarta_A

```
Yogyakarta_B#ping 202.147.192.1
```

mengirimkan ping menuju interface loopback0 Jakarta_B

Konfigurasi benar jika semua perintah ping berhasil.

3.3 RENCANA PENGUJIAN

Setelah perancangan sistem pada jaringan berarsitektur MPLS, rencana pengujian sistem yang akan dilakukan adalah melakukan *capture* paket data diantara semua *router* dengan menggunakan wireshark untuk menganalisa paket data tersebut, pengujian antara lain sebagai berikut :

1. Melakukan pengiriman paket data ICMP dari PC menuju Jakarta_B dan Jakarta_A.
2. Melakukan sniffing paket data dengan wireshark dari PC *client* virtualbox pada saat melakukan ping.
3. Melakukan pengujian apakah VPN terbentuk, apabila jaringan dapat saling terhubung berarti VPN terbentuk, sedang jika jaringan tidak berfungsi berarti VPN tidak terbentuk.
4. Melakukan uji pengamanan dengan menganalisa paket data yang jika telak terenkripsi berarti jaringan aman, namun jika tidak terenkripsi berarti jaringan tidak aman.

ISLAM
UNIVERSITAS
INDONESIA

BAB IV
HASIL DAN PEMBAHASAN

Bab ini menggambarkan tentang implementasi sistem pada penelitian ini. Implementasi sistem meliputi batasan implementasi sistem jaringan yang disimulasikan, pengujian sistem, analisis kinerja sistem, serta kelebihan dan kekurangan sistem.

4.1. Batasan Implementasi

Sistem yang dibangun adalah simulasi jaringan MPLS-VPN yang sederhana. Terdapat keterbatasan yang dimiliki sistem simulasi ini. Sistem berupa jaringan yang tidak terhubung ke Internet dan tidak meliputi server-server khusus, seperti DHCP server atau email server. Analisis implementasi hanya dilakukan sebatas protocol ICMP. Semua permasalahan tersebut terjadi karena keterbatasan perangkat keras yang digunakan dalam penelitian ini ternyata tidak mampu menangani simulasi yang dilakukan, ini berbanding terbalik dengan perangkat lunak yang mampu untuk mensimulasikan suatu sistem jaringan yang besar.

4.2. Tahap Proses Pembangunan Sistem

1. Analisis data

   Mengumpulkan berbagai data tentang teori jaringan MPLS dan cara untuk membangun simulasi sistemnya.

2. Perancangan sistem

   Menentukan topologi jaringan yang disimulasikan.

3. Pembangunan sistem

   Sistem dibangun dengan menggunakan software bantu GNS3, Virtualbox, dan Wireshark. Software GNS3 menggunakan perintah-perintah dari Cisco *Router* dan perintah dasar linux.

4. Pengujian sistem

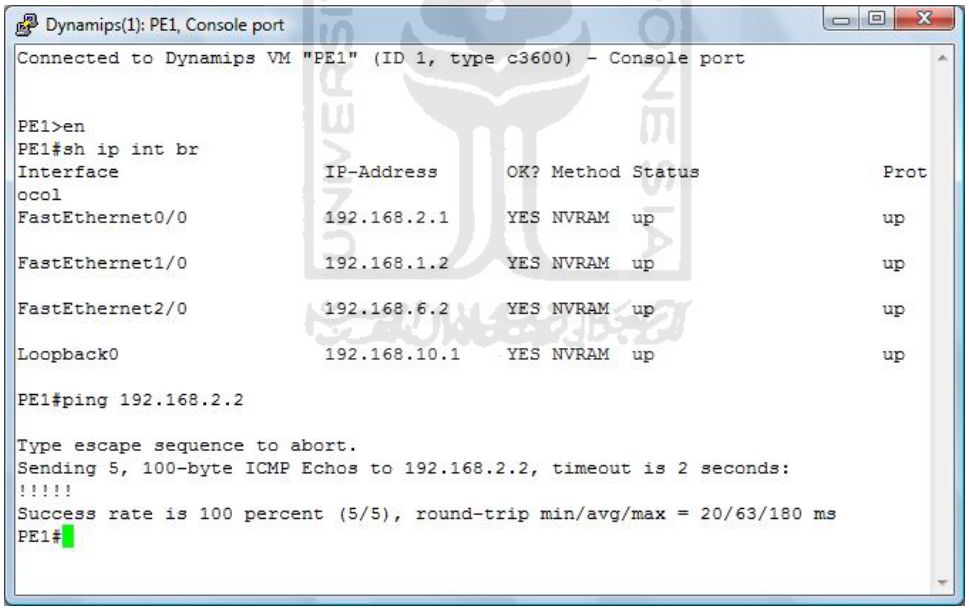
Pengujian sistem dilakukan dengan teknik sniffing menggunakan software Wireshark untuk menganalisis paket data.

4.3. Implementasi Sistem

Implementasi sistem jaringan MPLS-VPN merupakan hasil tahapan-tahapan pembangunan sistem.

4.3.1. Konfigurasi Interface

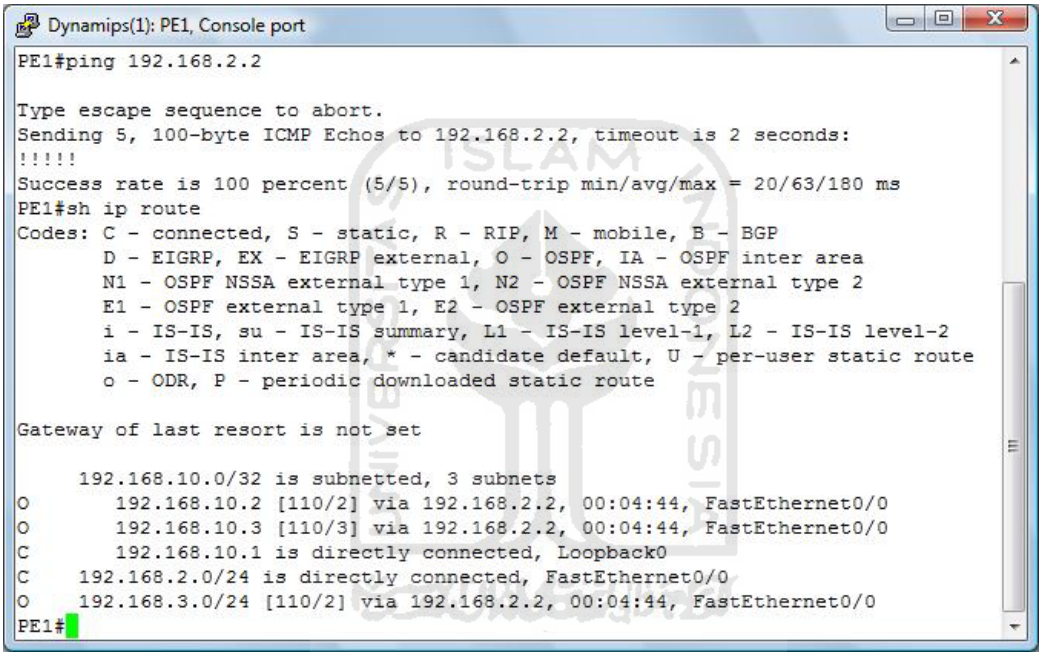
Konfigurasi ini dilakukan pertama kali pada semua *router* inti jaringan MPLS-VPN, yaitu *router* Core, PE1 dan PE2. Hasil konfigurasi memperlihatkan *router* telah saling terhubung dan dapat berkomunikasi. Gambar 3.1 adalah hasil konfigurasi dilihat dari jendela *console router* PE1.



Gambar 3.1 Konfigurasi interface PE1

4.3.2. Konfigurasi Routing OSPF

Konfigurasi routing OSPF ini bertujuan untuk menghubungkan interface loopback0 pada ketiga *router* inti jaringan MPLS-VPN. Routing OSPF ditandai dengan huruf "O" pada awal informasi interface-nya. Untuk mengetahui hasil dari konfigurasi routing OSPF dapat dilakukan dengan menjalankan perintah "sh ip route". Hasil konfigurasi OSPF selengkapnya di *router* PE1 dapat dilihat pada gambar 3.2

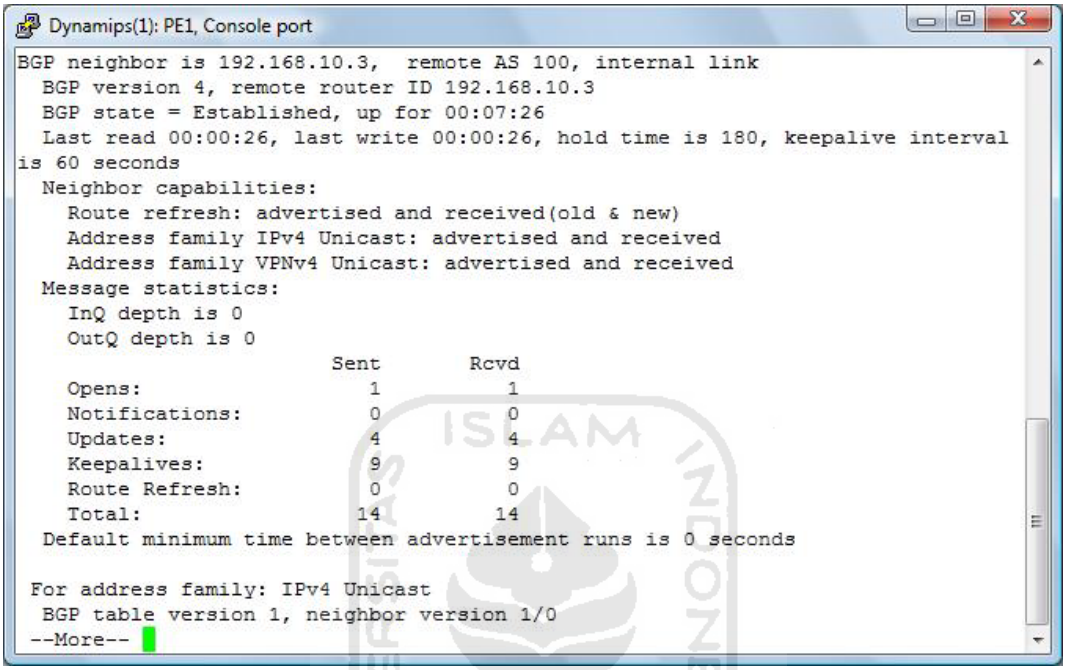


Gambar 3.2 Routing OSPF PE1

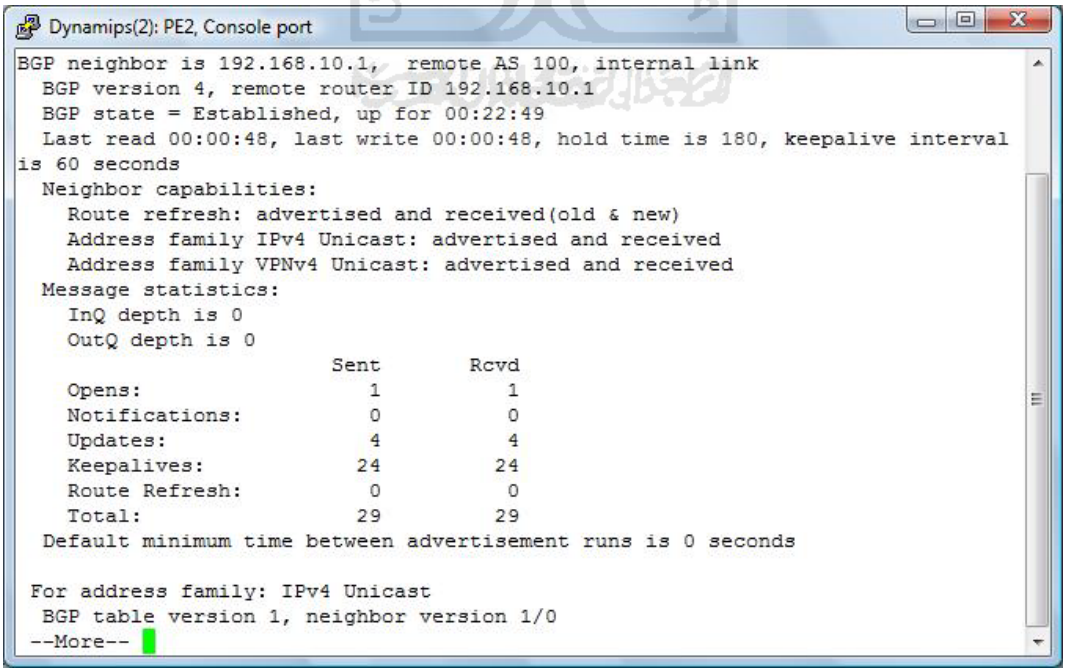
4.3.3. Konfigurasi Routing BGP

Konfigurasi ini digunakan untuk menghidupkan routing BGP antara *router* PE1 dan PE2 agar established artinya BGP antara keduanya telah saling terhubung sehingga status yang sebelum terhubung adalah *active* berubah menjadi established, hal ini diperlukan untuk konfigurasi selanjutnya yaitu konfigurasi MP-BGP yang merupakan ekstensi dari routing BGP. Untuk mengetahui hasil dari konfigurasi BGP ini dapat dilakukan dengan menjalankan perintah "sh ip bgp

neigh". Hasil dari konfigurasi BGP pada *router* PE dapat dilihat pada gambar 3.3 dan gambar 3.4.



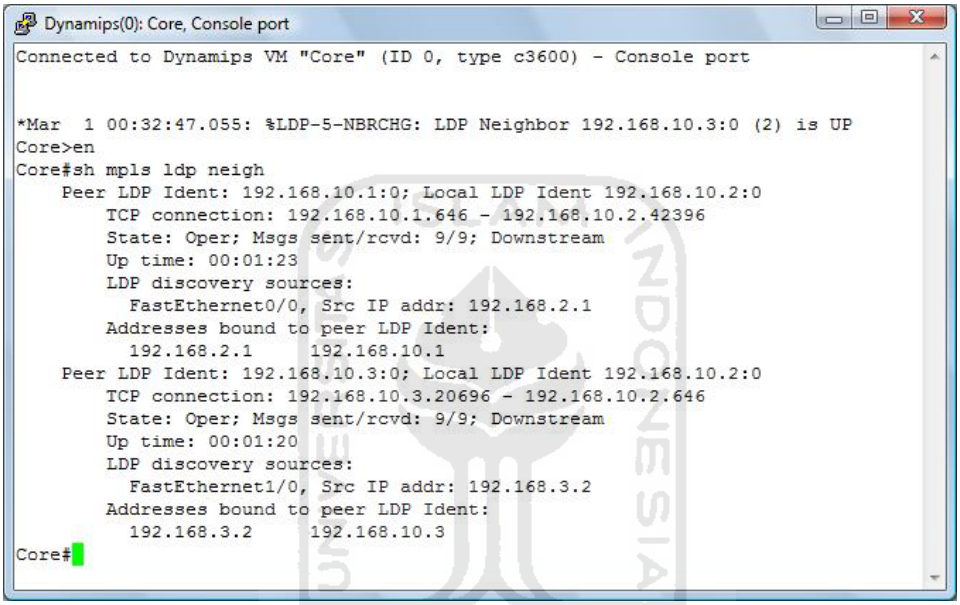
Gambar 3.3 Routing BGP Router PE1



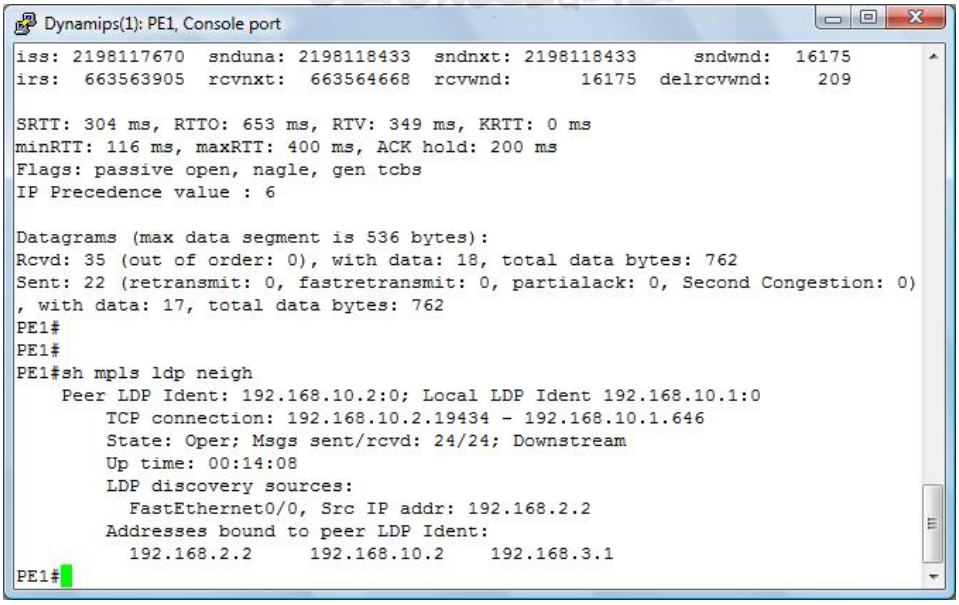
Gambar 3.4 Roting BGP Router PE2

4.3.4. Konfigurasi MPLS

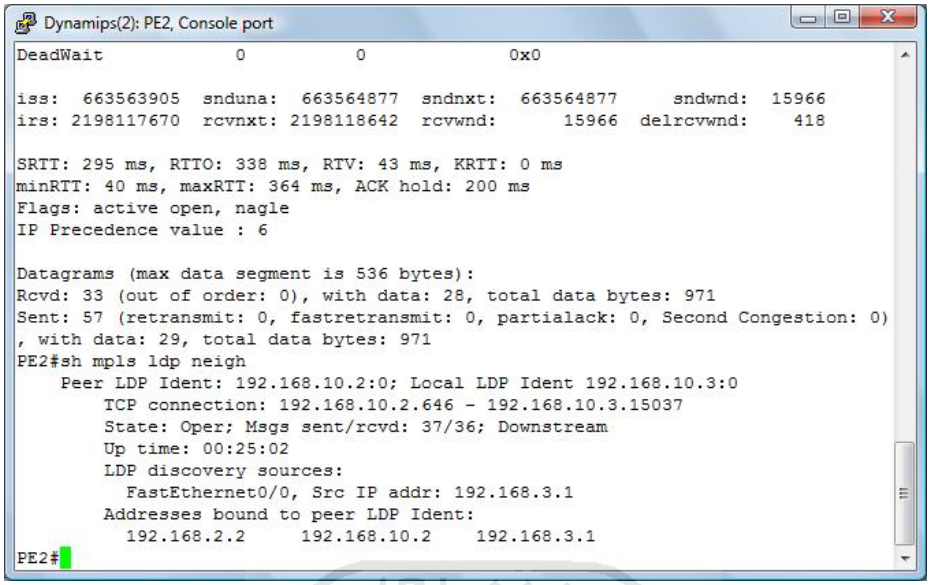
Konfigurasi MPLS dilakukan pada *router* PE1, Core dan PE2. Konfigurasi ini dilakukan untuk menukar informasi pelabelan MPLS di setiap *router* agar dapat saling mengenal sebagai MPLS ketika saling terhubung. Untuk mengetahui hasil dari konfigurasi ini dijalankan perintah “sh mpls ldp neigh”. Hasil konfigurasi MPLS di ketiga *router* tersebut dapat dilihat pada gambar 3.5, gambar 3.6 dan gambar 3.7.



Gambar 3.5 MPLS Router Core



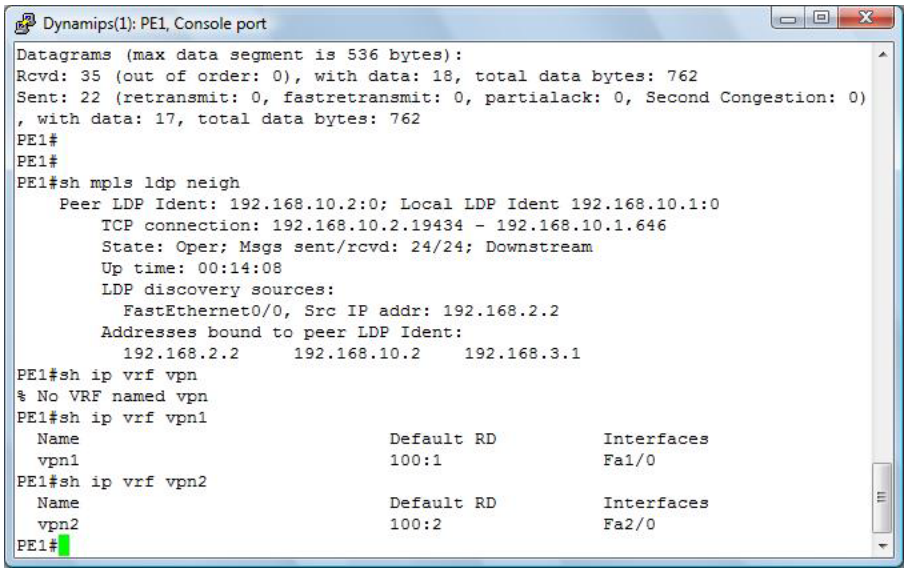
Gambar 3.6 MPLS Router PE1



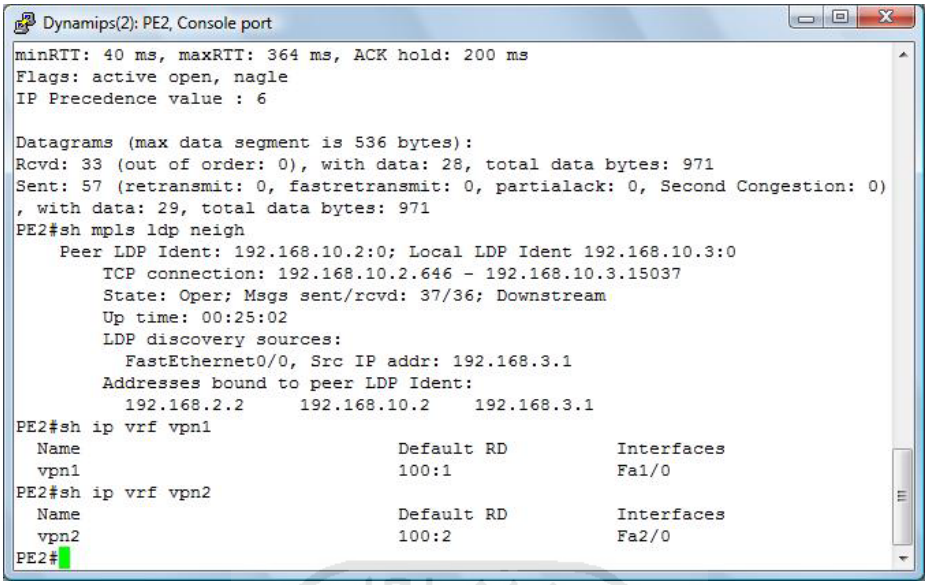
Gambar 3.7 MPLS Router PE2

4.3.5. Konfigurasi Router Virtual

Konfigurasi ini bertujuan membuat *router* virtual antara *router* PE1 dan PE2. *Router* virtual ini menggunakan sisa interface yang belum dikonfigurasi pada saat konfigurasi interface. Untuk mengetahui hasil dari konfigurasi *router* virtual dijalankan perintah “sh ip vrf vpn1”. Hasil dari konfigurasi dapat dilihat pada gambar 3.8 dan gambar 3.9.



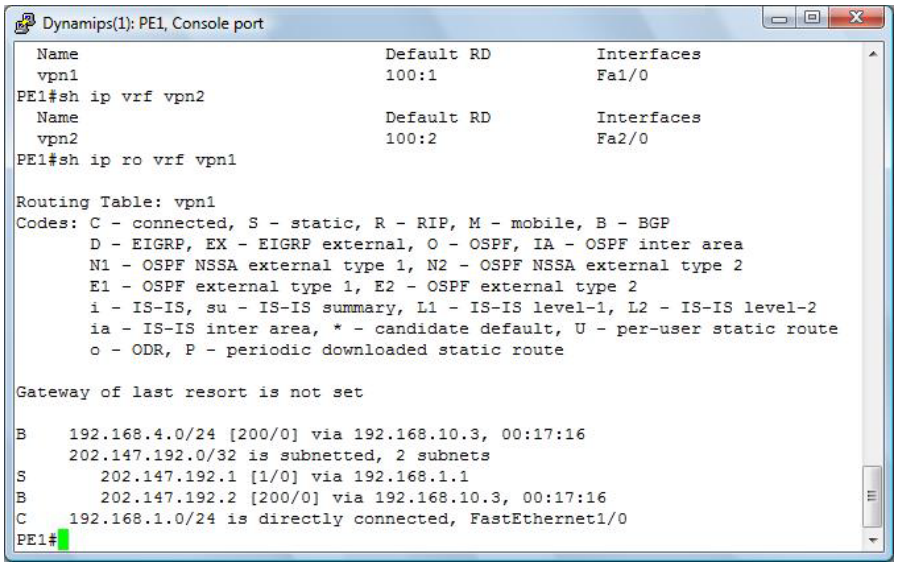
Gambar 3.8 Routing Virtual PE1



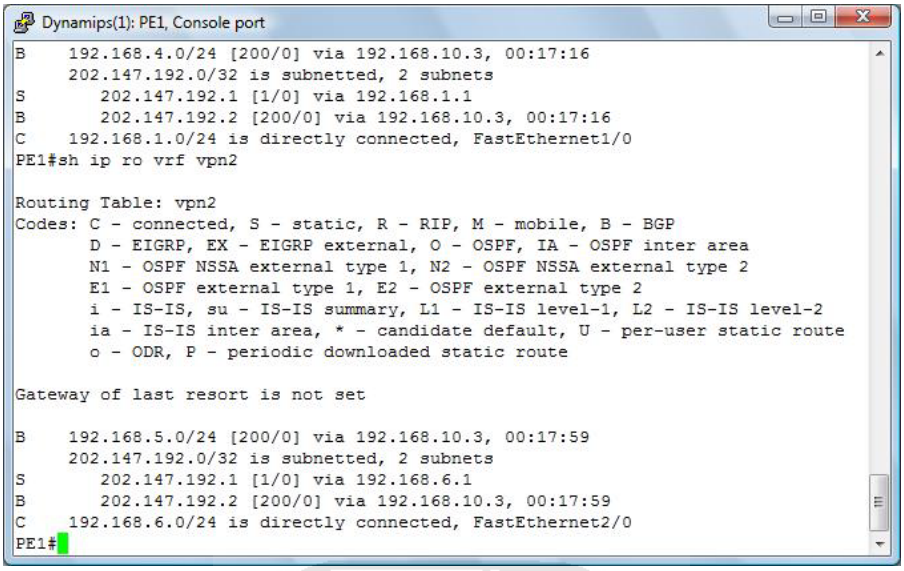
Gambar 3.9 Routing Virtual PE2

4.3.6. Konfigurasi Routing Static di Router Virtual

Konfigurasi ini bertujuan untuk memasukkan routing static pada *router* virtual di *router* PE yang akan digunakan oleh interface yang terhubung langsung dengan tiap *router* PE tersebut. Untuk mengetahui hasil dari konfigurasi ini dijalankan perintah “sh ip ro vrf von1”. Hasil dari konfigurasi yang ada di *router* PE1 dapat dilihat pada gambar 3.10 dan gambar 3.11.



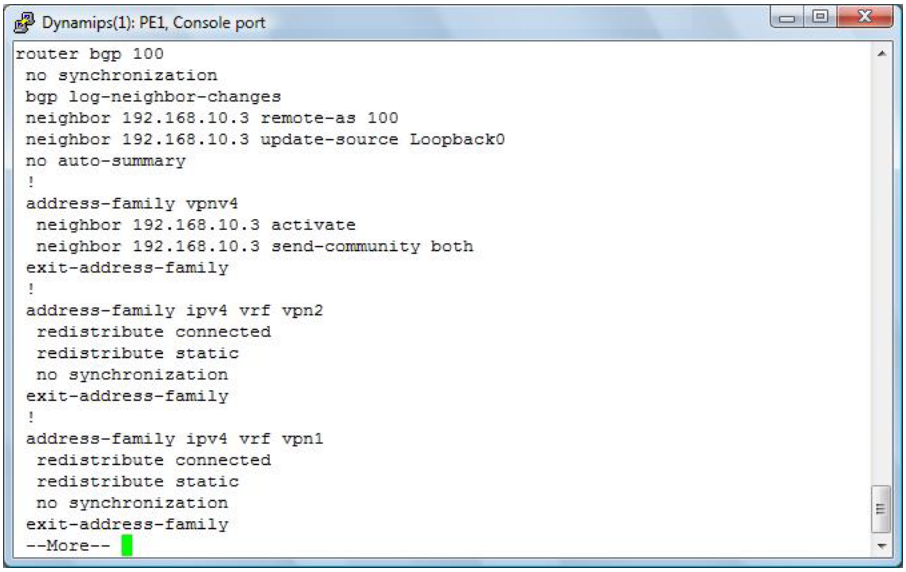
Gambar 3.10 Routing Static vpn1 di Router PE1



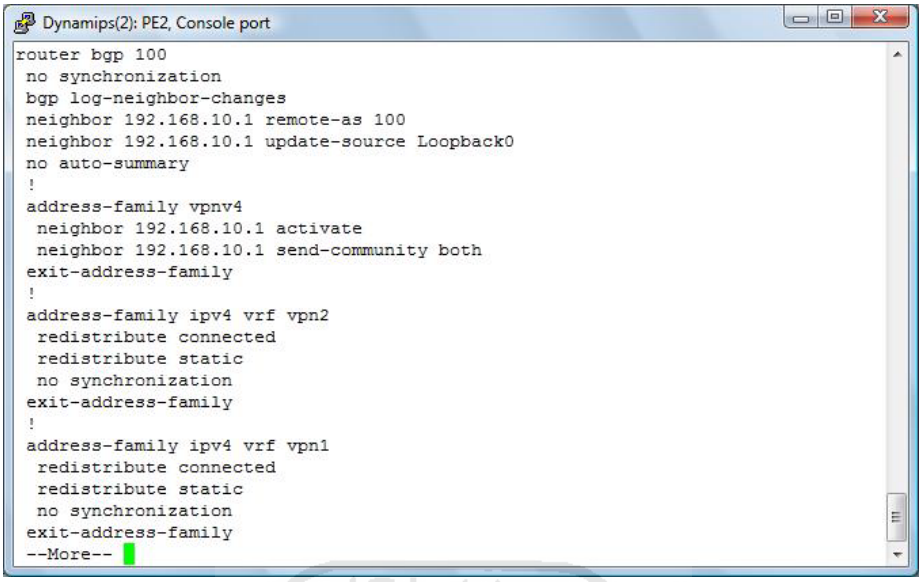
Gambar 3.11 Routing Static vpn2 di Router PE1

4.3.7. Konfigurasi Routing MP-BGP

Konfigurasi ini adalah lanjutan dari konfigurasi routing BGP. Routing MP-BGP merupakan sub-tunnel dari tunnel routing BGP yang berguna untuk membawa informasi table routing vrf vpn1 dan vpn2 pada *router* PE1 menuju PE2, dan demikian juga sebaliknya. Hasil dari konfigurasi MP-BGP dapat dilihat pada gambar 3.12 dan gambar 3.13.



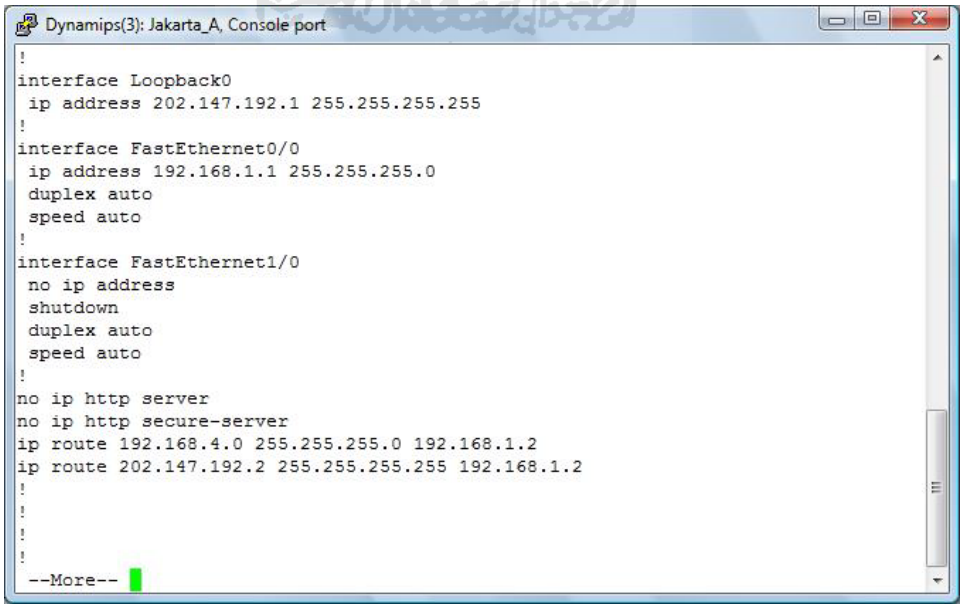
Gambar 3.12 Routing MP-BGP di Router PE1



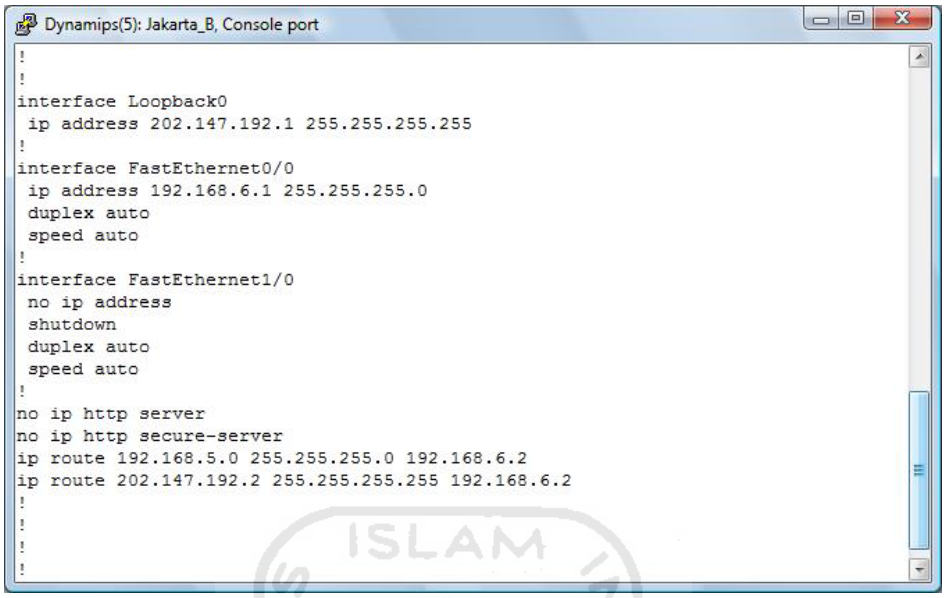
Gambar 3.13 Routing MP-BGP di Router PE2

4.3.8. Konfigurasi Router CE dan PC Client

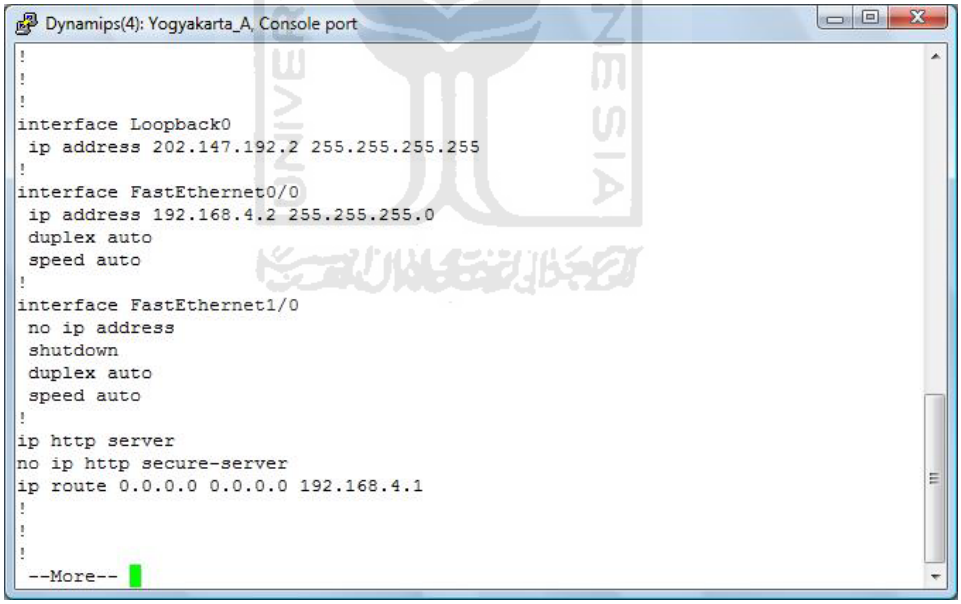
Konfigurasi ini dilakukan untuk menghubungkan *router-router* CE, yaitu Jakarta_A, Jakarta_B, Yogyakarta dan PC *client* pada jaringan MPLS yang telah dibangun. Hasil konfigurasi *router* dapat dilihat pada gambar 3.14, gambar 3.15, gambar 3.16 dan untuk konfigurasi PC dapat dilihat pada gambar 3.17.



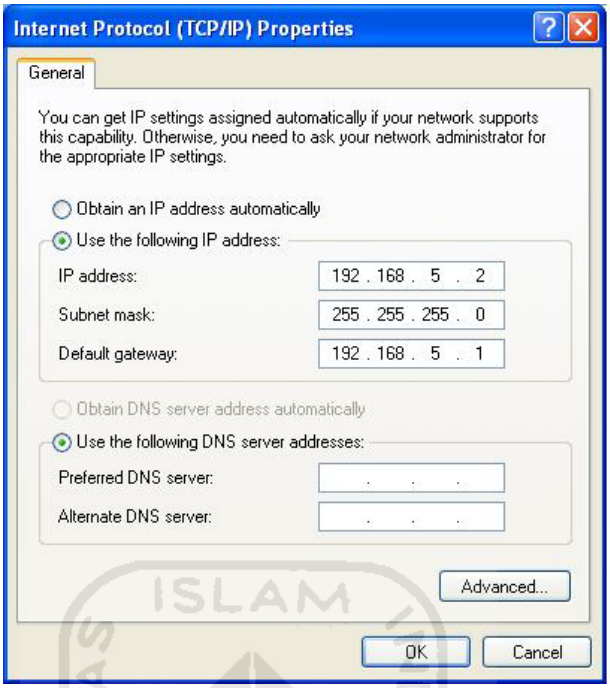
Gambar 3.14 Konfigurasi Jakarta_A



Gambar 3.15 Konfigurasi Jakarta_B



Gambar 3.16 Konfigurasi Yogyakarta_A



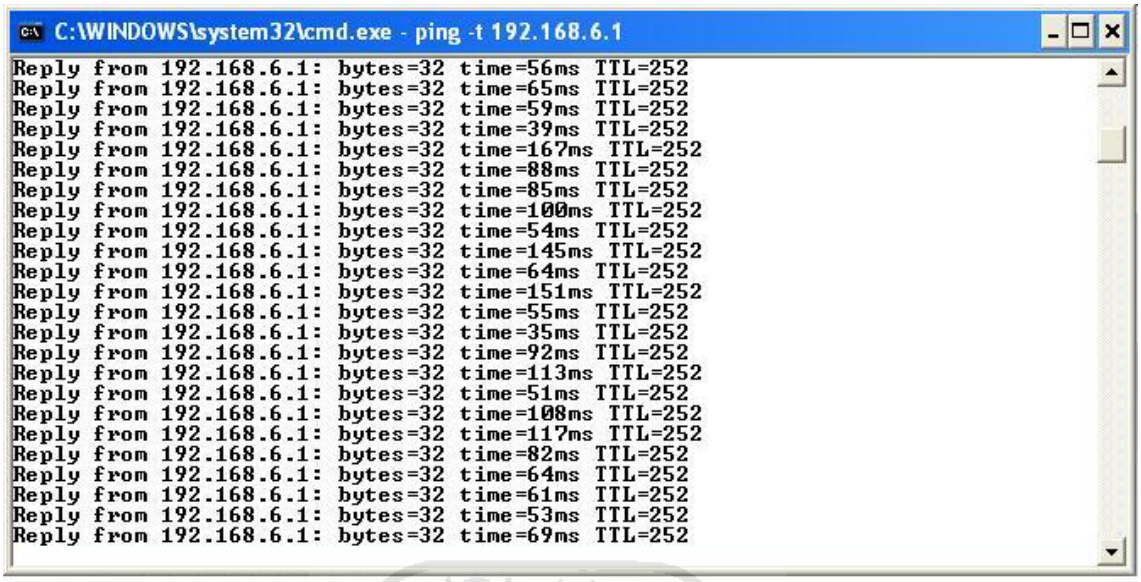
Gambar 3.17 Konfigurasi PC

4.4. Analisis

Pengujian yang dilakukan pada sistem jaringan MPLS ini berupa analisis sistem dari segi keamanannya dengan cara melakukan analisis paket data yang saling ditukar antara *router* menggunakan software Wireshark.

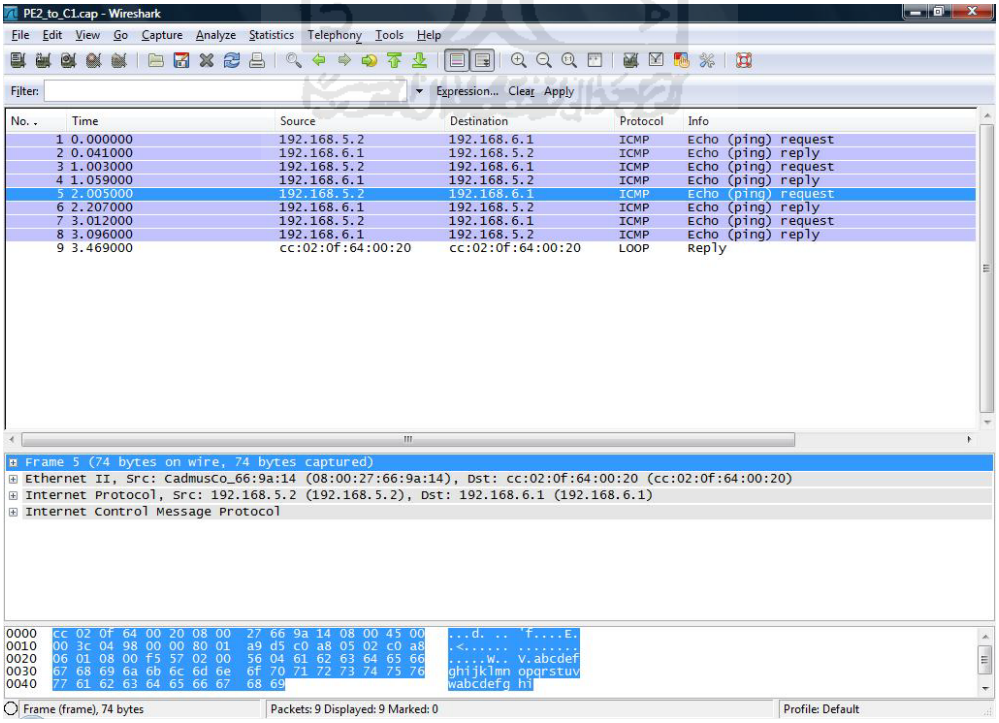
Skema pengujian dilakukan dengan menggunakan protokol ICMP yang dialirkan melalui PC menuju *router* CE yaitu Jakarta_B, kemudian akan dilakukan pengintaian paket data menggunakan Wireshark.

1. Mengalirkan data menggunakan protokol ICMP dari PC menuju Jakarta_B, yang dapat dilihat pada gambar 4.1

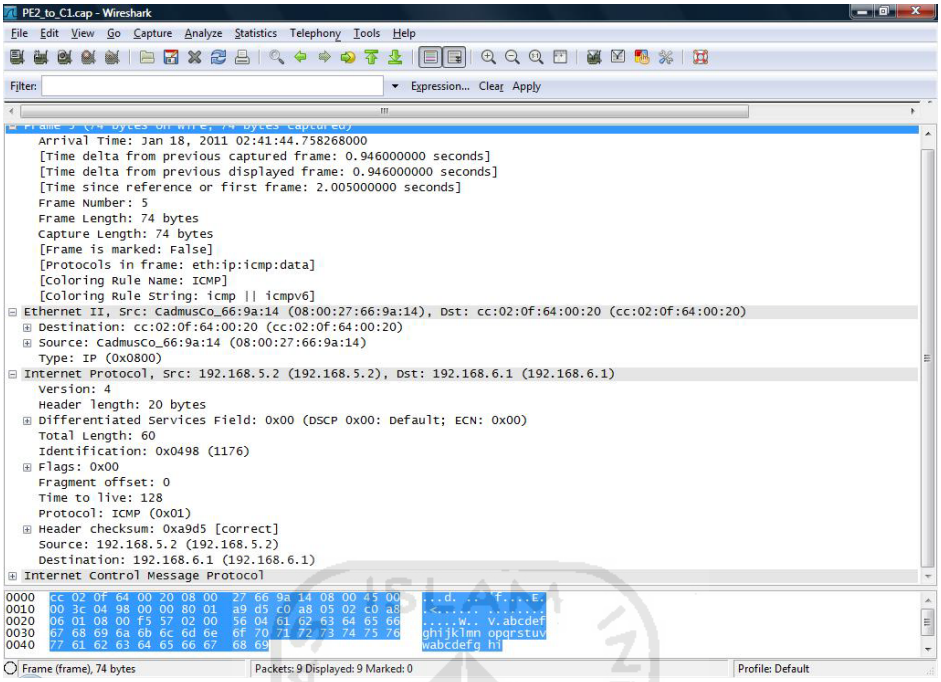


Gambar 4.1 ICMP paket dari PC ke Jakarta_B

2. Melakukan pengintaian paket data antara PC dan *router* PE2, hasil yang diperoleh wireshark dapat dilihat pada gambar 4.2 dan gambar 4.3.

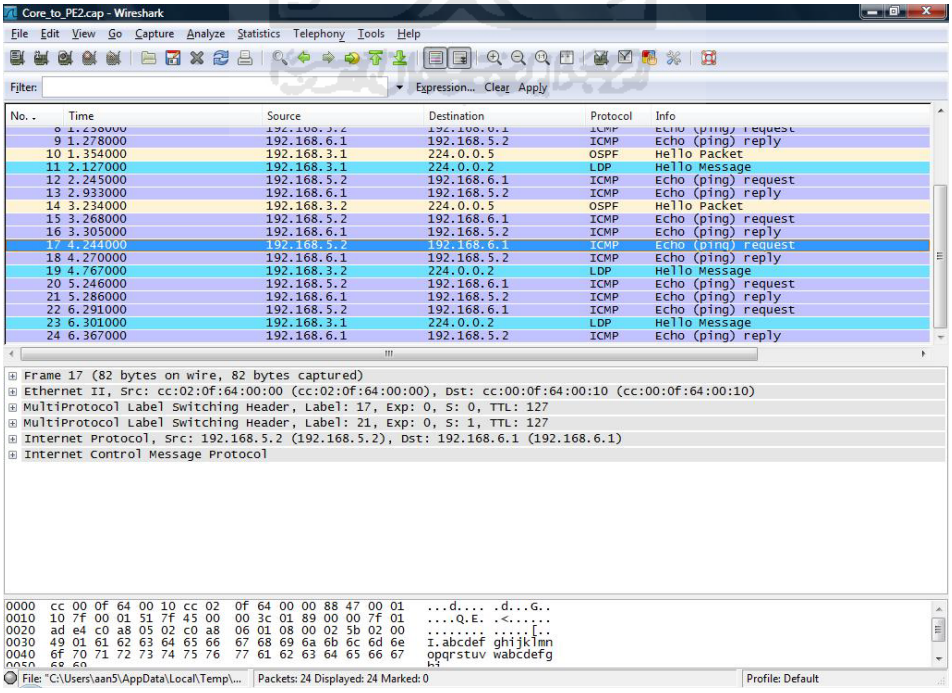


Gambar 4.2 Paket Data PE2 dan PC

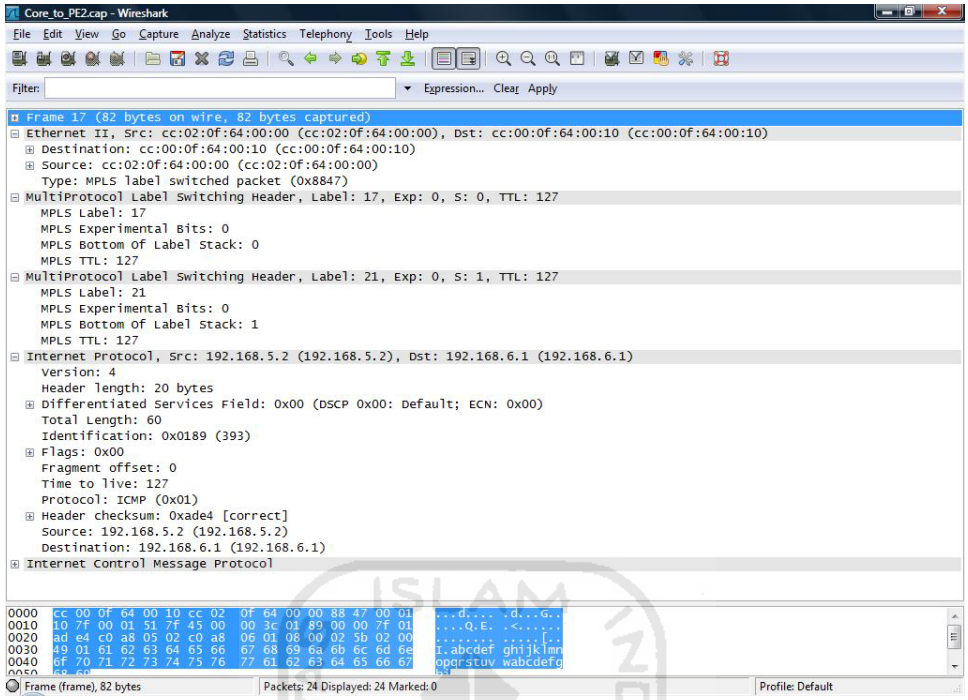


Gambar 4.3 Rincian Paket Data PE2 dan PC

3. Melakukan pengintaian paket data antara *router* Core dan *router* PE2 pada paket yang dikirimkan oleh PC, hasil yang diperoleh wireshark secara jelas dapat dilihat pada gambar 4.4 dan gambar 4.5.

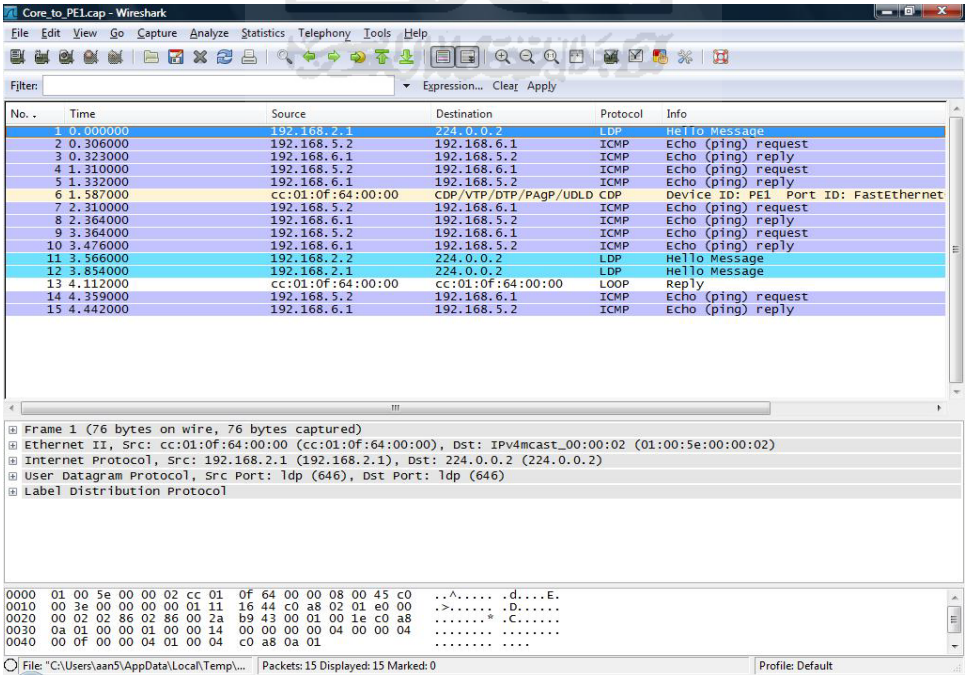


Gambar 4.4 Paket Data Core dan PE2

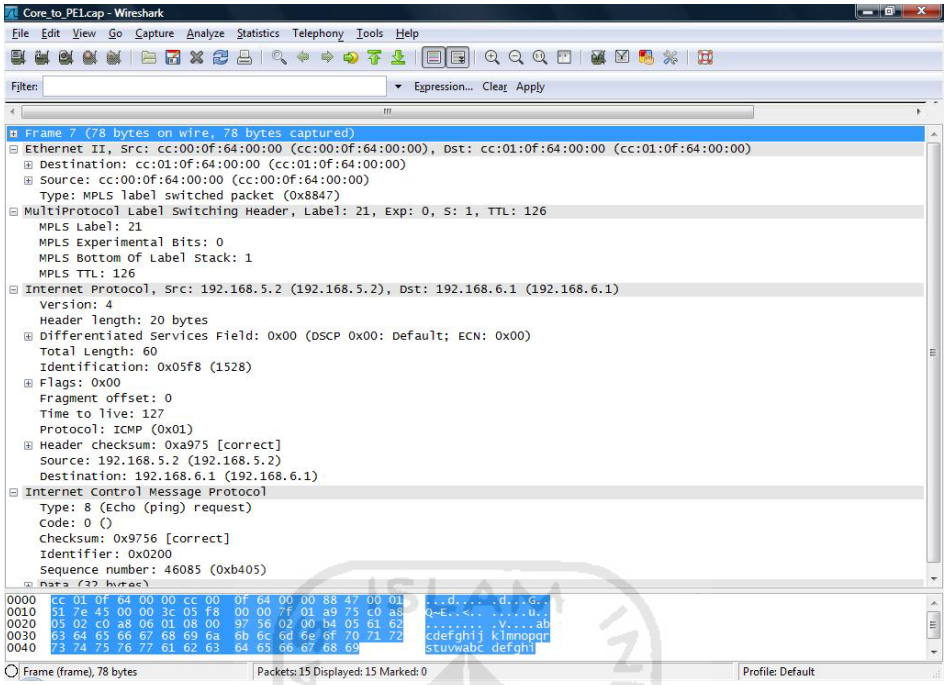


Gambar 4.5 Rincian Paket Data Core dan PE2

4. Melakukan pengintaian paket data antara *router* Core dan *router* PE1 pada paket yang dikirimkan oleh PC, hasil yang diperoleh wireshark dapat dilihat pada gambar 4.6 dan gambar 4.7.

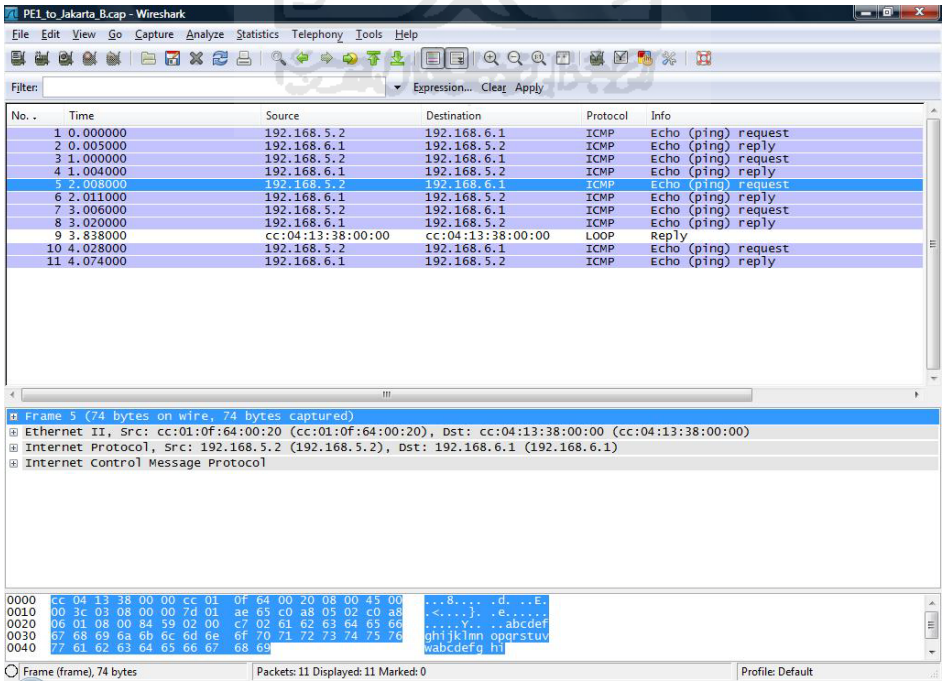


Gambar 4.6 Paket Data Core dan PE1

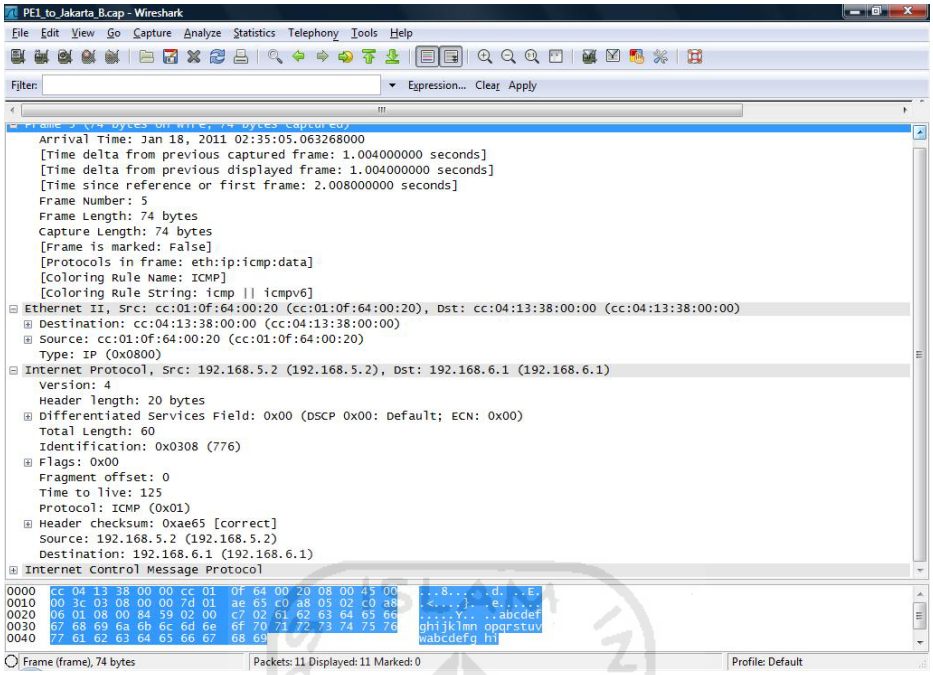


Gambar 4.7 Rincian Paket Data Core dan PE1

5. Melakukan pengintaian paket data antara *router* PE1 dan Jakarta_B pada paket yang dikirimkan oleh PC, hasil yang diperoleh wireshark dapat dilihat pada gambar 4.8 dan gambar 4.9.

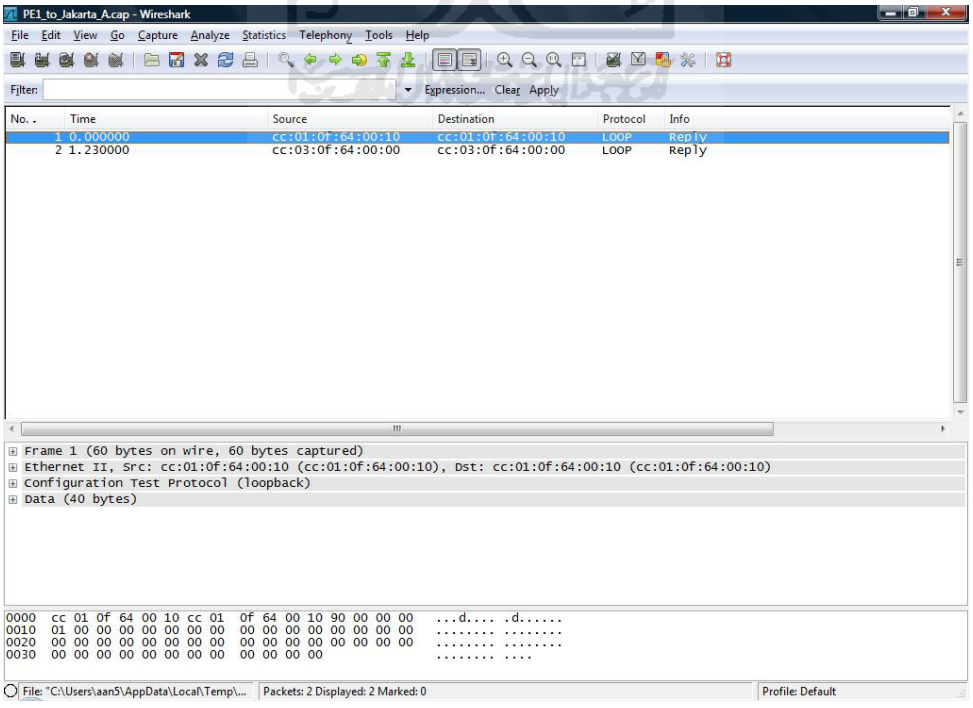


Gambar 4.8 Paket Data PE1 dan Jakarta_B



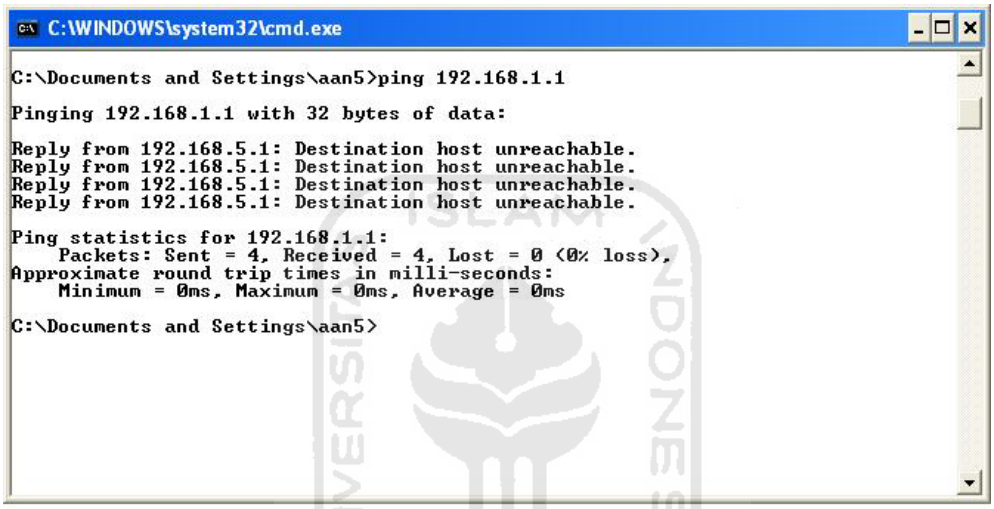
Gambar 4.9 Rincian Paket Data PE1 dan Jakarta_B

6. Melakukan pengintaian paket data antara *router* PE1 dan Jakarta_A pada paket yang dikrimkan oleh PC, hasil yang diperoleh wireshark dapat dilihat pada gambar 4.10

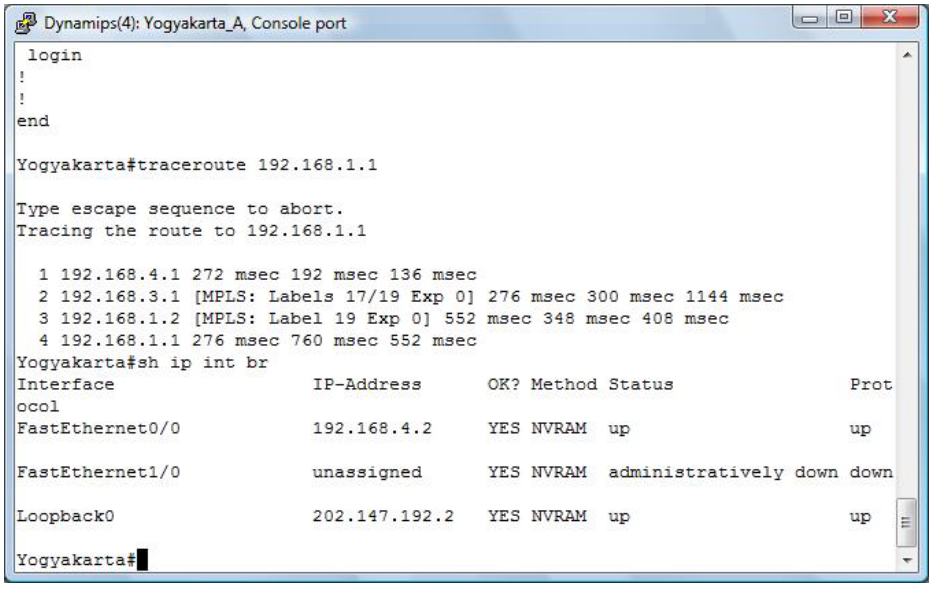


Gambar 4.10 Paket Data PE1 dan Jakarta_A

7. Untuk pengujian VPN, dilakukan uji koneksi antara PC menuju ke Jakarta_A dan melakukan pengujian traceroute dari Yogyakarta_A menuju Jakarta_A dan Jakarta_B, jika masih dapat saling terhubung antara PC menuju ke Jakarta_A dan terdapat jalur traceroute dari Yogyakarta_A menuju Jakarta_B berarti tunnel VPN gagal terbentuk, tetapi jika tidak terhubung berarti tunnel VPN berhasil terbentuk, hasil pengujian dapat dilihat pada gambar 4.11, gambar 4.12 dan gambar 4.13.

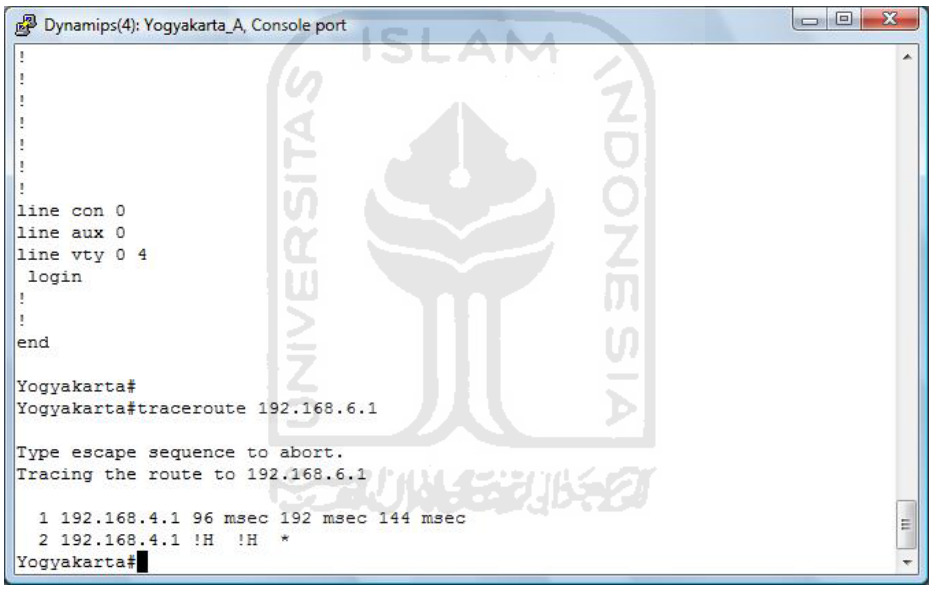


Gambar 4.11 Ping dari PC Menuju Jakarta_A



Gambar 4.12 Traceroute Yogyakarta_A Menuju Jakarta_A

Pada gambar 4.12 dan 4.13, dari hasil perintah *traceroute* yang dijalankan terdapat nilai dalam satuan *millisecond* setelah tulisan alamat IP, ini merupakan nilai latensi dari paket ICMP *Time-to-Live-Exceeded* yang merupakan *response* balasan dari paket *default User Datagram Protocol* (UDP) *traceroute* yang telah dikirimkan oleh Yogyakarta_A. Pada gambar 4.13, dari hasil perintah *traceroute* yang dijalankan terdapat informasi "!H" yang berarti *host* yang dituju *unreachable* dan tanda (*) yang berarti alamat IP tujuan tidak aktif atau tidak melakukan *response* terhadap paket *traceroute* yang telah dikirimkan oleh Yogyakarta_A.



Gambar 4.13 Traceroute Yogyakarta_A Menuju Jakarta_B

Dari hasil pengujian, dapat dilakukan analisis terhadap hasil pengujian tersebut. Berikut adalah uraian hasil analisis sistem keamanan pada jaringan MPLS :

1. Aliran data dari PC menuju *router* PE2 berhasil dilakukan dan pada aliran data ini tidak ditemukan paket data MPLS.
2. Aliran data dari *router* PE2 menuju *router* Core berhasil dilakukan dan pada aliran data ini telah ditemukan adanya paket data MPLS.

3. Aliran data dari *router* Core menuju *router* PE1 berhasil dilakukan dan pada aliran data ini dapat ditemukan paket data MPLS.
4. Aliran data dari *router* PE1 menuju Jakarta_B berhasil dilakukan dan pada aliran data ini tidak ditemukan paket data MPLS.
5. Aliran data dari *router* PE1 menuju Jakarta_A tidak ditemukan.
6. Tidak ditemukan enkripsi oleh wireshark pada semua paket data yang dikirimkan oleh PC ataupun pada paket data yang diteruskan oleh *router*.

4.5 ANALISIS SISTEM

Setelah melakukan konfigurasi jaringan MPLS ini dapat ditarik beberapa analisis mengenai sistem yang telah di bangun, yaitu:

1. Jaringan berarsitektur MPLS terdiri dari routing dynamic BGP,OSPF, dan MP-BGP.
2. Jaringan MPLS juga menggunakan routing static untuk menghubungkan virtual *router* dengan *router* CE nya.
3. Jaringan MPLS menggunakan virtual *router* VRF sebagai *router virtual*.
4. MPLS hanya dapat diterapkan pada *router-router* tertentu seperti *router* Cisco yang bersifat *manageable*.

4.6 Analisis Kelebihan dan Kekurangan Sistem

Bagian ini membahas kelebihan dan kelemahan keamanan sistem jaringan MPLS yang telah dibuat.

a. Kelebihan sistem keamanan jaringan berasitektur MPLS adalah:

1. Memiliki *router virtual* pada *router* MPLS yang langsung terhubung dengan *router client* yang berfungsi *memforward* paket langsung ke *router client* yang menjadi tujuan paket data sesuai *routing target*-nya sehingga paket data tetap aman ketika terlepas dari enkapsulasi MPLS.

2. Pada *router* Jakarta A dan B memiliki interface loopback0 yang beralamat IP sama, namun MPLS dapat melakukan pemilihan paket data yang harus langsung dikirim sesuai tujuan.
3. *Tunneling* VPN yang menciptakan lorong khusus untuk menghubungkan antar *router* sehingga *router* yang saling terhubung seakan-akan terdapat pada satu jaringan lokal yang sama dengan memanfaatkan jaringan publik. Hal ini memberikan keamanan pada paket data karena paket data tersebut tidak akan dibocorkan ke luar *tunneling* VPN yang tidak terdefinisikan.

b. Kelemahan sistem keamanan jaringan berasitektur MPLS adalah :

1. Jika *router* MPLS dapat disusupi, maka paket data akan mudah untuk dicuri.
2. Enkripsi MPLS tidak dilakukan sebelum paket datang ke *router* PE2 dari PC dan setelah paket dilepas dari *router* PE1, hal ini memberikan celah terjadinya pembacaan pada paket data tersebut.

BAB V
SIMPULAN DAN SARAN

Simpulan

Berdasarkan hasil penelitian dapat ditarik beberapa kesimpulan sebagai berikut:

a. Suatu sistem jaringan MPLS dapat disimulasikan dengan perangkat lunak GNS3 sebagai simulator *router* Cisco dan Virtualbox sebagai simulator PC *client*.
b. MPLS dapat diintegrasikan dengan VPN menjadi sebuah jaringan yang memiliki sistem keamanan yang cukup baik karena memiliki *tunneling* sendiri pada jaringan yang bersifat publik.
c. Jika keterbatasan perangkat keras yang ada dapat diatasi, sistem jaringan yang lebih kompleks akan mampu disimulasikan untuk penelitian sistem keamanan yang lebih baik.
d. Paket data MPLS hanya ter-enkapsulasi dan tidak ter-enkripsi.

Saran

Berdasarkan kekurangan dan keterbatasan yang muncul dalam penelitian sistem keamanan jaringan yang berarsitektur MPLS ini, maka saran untuk pengembangan penelitian di masa yang akan datang adalah sebagai berikut:

a. Penelitian Analisis sistem keamanan ini dapat lebih dikembangkan lagi dengan menggabungkan MPLS dan IPSec, atau sistem keamanan jaringan lainnya, karena pada penelitian ini tidak digunakan sistem keamanan tersebut.
b. Teknik analisis yang digunakan untuk pengujian keamanan tidak hanya dilakukan dengan *sniffing* pasif tapi juga dengan *sniffing* aktif.
c. Ditambahkan *server-server* lain pada simulasi jaringan MPLS agar analisis keamanan tidak terbatas hanya dilakukan pada protokol ICMP.
d. Dapat ditambahkan enkripsi agar paket data tidak hanya dienkapsulasi saja oleh MPLS.

DAFTAR PUSTAKA

Cisco. 2002. *Advanced MPLS Design and Implementation.* Indianapolis : Cisco Press

Cisco. 2007. *CCNA Exploration 4.0 Network Fundamental.* California : Cisco System, Inc

Cisco. 2007. *CCNA Exploration 4.0 Routing Protocol Concepts.* California : Cisco System, Inc

Ilyas, Hernandi. 2011. *Teknologi Switching*. (On-line) Available at http://www.scribd.com/rokhmatf.

Infusion, IP. 2001. *MPLS-VPN*. San Jose : IP Infusion, Inc

Irianto, Antonius. 2011. *Model Jaringan 7 Osi Layer*. (On-line) Available at http://irianto.staff.gunadarma.ac.id/Downloads/files/16422/MODEL+JARINGAN+7+OSI+LAYER.pdf.

Komputer, Wahana. 2009. *Langkah Mudah Administrasi Jaringan Menggunakan Linux Ubuntu 9.* Semarang : Andi Offset

Netlab, UI. 2010. *Network Analysis Tool, Application Layer Protocol, dan Transport Layer Protocol* (On-line) Available at http://www.ee.ui.ac.id/

Saputro, Joko. 2010. *Praktikum CCNA (Cisco Certified Network Associate) di Komputer Sendiri Menggunakan GNS3*. Jakarta : Media Kita

Wastuwibowo, Kuncoro. 2003. *Pengantar MPLS*. Bandung : Ilmu Komputer.Com

ISLAM
UNIVERSITAS
INDONESIA