UNIVERSITAS INDONESIA

IMPLEMENTASI DAN ANALISA PERBANDINGAN QOS PADA JARINGAN VPN BERBASIS MPLS MENGGUNAKAN ROUTING PROTOKOL RIPV2, EIGRP DAN OSPF TERHADAP TUNNELING IPSEC UNTUK LAYANAN IP-BASED VIDEO CONFERENCE

SKRIPSI

ROSYIDINA SAFITRI
0806366270

FAKULTAS TEKNIK
PROGRAM STUDI TEKNIK ELEKTRO
UNIVERSITAS INDONESIA
DEPOK
JUNI 2010

UNIVERSITAS INDONESIA

IMPLEMENTASI DAN ANALISA PERBANDINGAN QOS PADA JARINGAN VPN BERBASIS MPLS MENGGUNAKAN ROUTING PROTOKOL RIPV2, EIGRP DAN OSPF TERHADAP TUNNELING IPSEC UNTUK LAYANAN IP-BASED VIDEO CONFERENCE

SKRIPSI

Diajukan sebagai salah satu syarat untuk memperoleh gelar Sarjana Teknik

ROSYIDINA SAFITRI
0806366270

FAKULTAS TEKNIK
PROGRAM STUDI TEKNIK ELEKTRO
UNIVERSITAS INDONESIA
DEPOK
JUNI 2010

HALAMAN PERNYATAAN ORISINALITAS

Skripsi ini adalah hasil karya saya sendiri,

dan semua sumber baik yang dikutip maupun dirujuk

telah saya nyatakan dengan benar.

Nama : Rosyidina Safitri

NPM : 08 06 36 62 70

Tanda Tangan :

Tanggal : 1 Juli 2010

HALAMAN PENGESAHAN

Skripsi ini diajukan oleh :
Nama : Rosyidina Safitri
NPM : 0806366270
Program Studi : Teknik Elektro
Judul Skripsi : Implementasi dan Analisa Perbandingan *QoS* pada Jaringan *VPN* berbasis *MPLS* Menggunakan *Routing Protocol RIPv2, EIGRP*, dan *OSPF* terhadap *Tunneling IPSec* untuk Layanan *IP-based Video Conference*

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

DEWAN PENGUJI

Pembimbing : Muhammad Salman ST, MIT ( )

Penguji : Prof. Dr. Ir. Riri Fitri Sari, M.Sc, MM ( )

Penguji : Prof. Dr. Ing. Ir. Kalamullah Ramli, M.Eng ( )

Ditetapkan di : Depok

Tanggal : 1 Juli 2010

KATA PENGANTAR

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

1. Muhammad Salman ST, MIT selaku dosen pembimbing yang telah menyediakan waktu, tenaga, dan pikiran untuk mengarahkan saya dalam penyusunan skripsi ini.
2. Orang tua dan keluarga saya yang telah memberikan bantuan dukungan secara moral.
3. Sahabat yang telah banyak membantu saya dalam menyelesaikan skripsi ini.

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

Depok, 1 Juli 2010

Penulis

HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS

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

Nama	: Rosyidina Safitri
NPM	: 0806366270
Program Studi	: Teknik Elektro
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 :

IMPLEMENTASI DAN ANALISA PERBANDINGAN *QOS* PADA JARINGAN *VPN* BERBASIS *MPLS* MENGGUNAKAN *ROUTING PROTOCOL RIPv2, EIGRP,* DAN *OSPF* TERHADAP *TUNNELING IPSEC* UNTUK LAYANAN *IP-BASED VIDEO CONFERENCE*

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 skripsi 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 : 1 Juli 2010

Yang menyatakan

(Rosyidina Safitri)

ABSTRAK

Nama : RosyidinaSafitri
Program Studi : Teknik Elektro
Judul : Implementasi dan Analisa Perbandingan *QoS* pada Jaringan *VPN* berbasis *MPLS* Menggunakan *Routing Protocol RIPv2, EIGRP*, dan *OSPF* terhadap *Tunneling IPSec* untuk Layanan *IP-based Video Conference*

Berkembangnya konvergensi internet dan telekomunikasi memberikan dukungan penuh terhadap keamanan data dan peningkatan kinerja jaringan. IETF menstandarkan MPLS sebagai pengembangan dari teknologi VPN. MPLS dapat menyederhanakan proses *routing* yang menjadi beban router, mengoptimalkan pemilihan jalur melalui kemampuan manajemen *class of service* dan *traffic engineering*. Untuk mendukung optimasi pemilihan jalur, *routing protokol* mempunyai peran yang fundamental didalam jaringan. Pemilihan routing protokol yang tepat diperlukan agar jaringan optimal dan efisien, serta dapat mengatasi situasi routing yang kompleks secara cepat dan akurat. *IPSec* diimplementasikan pada *end-to-end* router untuk memberikan proteksi pada lapisan jaringan dengan merancang mekanisme keamanan *kriptografi*. Implementasi tunnel IPSec pada jaringan MPLS yang dijalankan pada tiga routing protokol yang berbeda yaitu *RIPv2*, *EIGRP* dan *OSPF* dengan aplikasi *video conference* sebagai data pengujian menunjukkan bahwa perubahan kualitas pada trafik video lebih besar daripada suara, dimana untuk trafik video RIPv2 memiliki delay terbesar, sedangkan delay EIGRP dan OSPF relatif sama. Setelah IPSec di implementasikan, terjadi kenaikan delay secara signifikan pada OSPF sebesar 101%, sedangkan pada RIPv2 dan EIGRP hanya sekitar 8%. Parameter pengujian lainnya seperti *jitter, throughput* dan *packet loss* lebih banyak dipengaruhi oleh delay yang terukur, sehingga diperoleh kesimpulan bahwa EIGRP adalah routing protokol yang memiliki kinerja paling bagus dilihat dari parameter jaringan yang terukur, didukung dengan nilai MOS dari responden dan paling efektif untuk diimplementasikan pada jaringan MPLS dengan tunnel IPSec dalam skala network yang kecil dan bandwidth yang terbatas.

Kata kunci :
MPLS, VPN, IPSec, kriptografi*, video conference*, *RIPv2, EIGRP, OSPF, delay, jitter, packet loss, throughput*

ABSTRACT

Name : Rosyidina Safitri
Study Program: Electrical Engineering
Title : Implementation and Comparison Analysis of QoS in MPLS-based VPN Network Using RIPv2, EIGRP, and OSPF Routing Protocols over IPSec Tunneling for IP-based Video Conference Service

The growth of Internet convergence and telecommunication provide full support for data security and improvement in network performance. IETF standardize MPLS as part of VPN techology development. MPLS can simplify routing process which became a load in router, optimizing route selection through the capability of class of service and traffic engineering management. To support optimization in route selection, routing protocols have fundamental role in the network. Routing protocol selection is required for network become optimum and efficient, also can handle complex routing situation more fast and accurate. IPSec is implemented to *end-to-end* router to provide protection at the network layer by designing cryptography mechanism. IPSec tunnel implementation over MPLS network that running at three different routing protocols RIPv2, EIGRP and OSPF using *video conference* application as testing data, shows that the change of quality in video traffic is bigger than voice. Video traffic RIPv2 has the largest delay, while EIGRP and OSPF delay relatively the same. But after IPSec is implemented, delay significantly increase in OSPF by 101% while RIPv2 and EIGRP increase about 8%. Other testing parameters such as *jitter, throughput* dan *packet loss* are more influenced by measured delay, thus concluded that EIGRP is the best routing protocol in performance from measured parameters, supported with MOS value from respondents and the most effective to be implemented in MPLS network with IPSec tunnel in small scale network and limited bandwidth.

Key words :
MPLS, VPN, IPSec, cryptography, video conference, RIPv2, EIGRP, OSPF, delay, jitter, packet loss, throughput

DAFTAR ISI

DAFTAR GAMBAR

DAFTAR TABEL

BAB 1
PENDAHULUAN

1.1 LATAR BELAKANG

Kebutuhan akan komunikasi data yang terintegrasi saat ini sudah menjadi kebutuhan utama bagi sebuah institusi atau perusahaan, apalagi saat ini banyak perusahaan cenderung mempunyai banyak cabang yang tersebar di lokasi yang berjauhan, belum lagi seorang pegawai memerlukan akses ke file-file, email dan database di kantor pusat yang memerlukan koneksi langsung ke server. Kegiatan tersebut bisa menjadi sangat mahal dan memerlukan *hardware* dan dukungan teknis yang rumit.

Berkembangnya konvergensi internet dan telekomunikasi, dengan aplikasi didalamnya yang kian tergantung pada ketersediaan *bandwidth* yang besar, dengan pengaturan *QoS*-nya membutuhkan jaringan dan elemen didalamnya yang memberi dukungan penuh terhadap keamanan data dan peningkatan kinerja jaringan. Maka dibutuhkan teknologi pengiriman data yang tidak hanya memudahkan *routing* dan *discovery* lintasan terbaik, namun juga dapat memberikan keamanan dalam melakukan komunikasi data.

IETF menstandarkan solusi *Multi Protocol Label Switching* (*MPLS*) sebagai pengembangan dari teknologi *Virtual Private Network* (*VPN)* untuk meningkatkan kinerja *forwarding* dan kecerdasan *traffic engineering* pada jaringan *packet-based*. Teknologi ini dapat menyederhanakan proses *routing* yang menjadi beban router karena harus menganalisa setiap *header IP* yang masuk, serta mengoptimalkan pemilihan *path* melalui kemampuan manajemen *class of service* dan *traffic engineering*.

Dalam optimasi pemilihan *path*, *routing* protokol mempunyai peran yang sangat fundamental bagi jaringan, karena dengan *routing protocol* router mengetahui kemana data harus dikirim, dan sebuah *autonomous system* memerlukan *routing protocol* yang dapat berkonvergensi dengan cepat dan efisien. Pemilihan *routing* protokol yang tepat akan memperkuat manajemen lalu lintas data karena *routing* protokol tidak hanya didesain untuk mengubah ke rute *backup* bila rute utama tidak berhasil, namun juga didesain untuk menentukan rute

mana yang terbaik untuk mencapai tujuan dan mengatasi situasi *routing* yang kompleks secara cepat dan akurat.

Jaringan *MPLS* tidak cukup aman bila diterapkan tanpa adanya mekanisme keamanan yang dapat melindungi data yang dikirimkan melalui *public network*. Oleh sebab itu *IPSec* diimplementasikan pada setiap *end-to-end* router untuk menjamin keamanan data dan ketertutupan transfer data dari akses ilegal yang tidak diinginkan. *IPSec* melakukan proteksi lapisan jaringan dengan merancang mekanisme keamanan pengacakan (kriptografi). Sehingga sebuah perusahaan yang menggunakan layanan *MPLS* dari suatu penyedia jasa dan mengimplementasikan *IPSec* disetiap routernya dapat membuat jalur aman yang didukung dengan manajemen *Quality of Service (QoS)* yang baik dalam melakukan komunikasi dengan kantor-kantor cabangnya (*VPN Intranet*) dan mengembangkan komunikasi dengan partner bisnisnya (*VPN Ekstranet*).

Pada tugas akhir ini dilakukan implementasi dan analisa perbandingan pengaruh kinerja *routing* protokol *RIPv2, EIGRP* dan *OSPF* terhadap implementasi *IPSec* pada jaringan *MPLS*. Data yang diujikan berupa trafik *video conference,* dikarenakan *video conference* merupakan aplikasi *realtime* yang sensitif terhadap *delay* dan *packet drop*. Diharapkan dengan data *video conference* dapat diketahui *routing protocol* yang paling optimal dan paling stabil untuk diimplementasikan dalam jaringan *MPLS* dengan tunnel *IPSec*.

1.2 Perumusan Masalah

Permasalahan pada tugas akhir ini adalah untuk mengetahui pengaruh penentuan/pemilihan rute oleh *routing* protokol *RIPv2, EIGRP* dan *OSPF* terhadap implementasi *IPSec* pada jaringan *MPLS*. Ketika sebuah jaringan *MPLS* diterapkan *IPSec* disetiap *end-to-end* router dengan menggunakan *routing protocol* yang berbeda, yaitu *RIPv2, EIGRP* dan *OSPF* dalam kondisi yang berbeda, yaitu pada saat kondisi jaringan stabil dan tidak stabil dengan *bandwidth* yang minimum, maka *routing* protokol mana yang paling optimal, mempunyai kinerja paling baik dan paling efektif dilihat dari parameter-parameter jaringan yang terukur dan nilai *MOS* yang diperoleh dari responden.

1.3 Tujuan Penulisan

Penyusunan tugas akhir ini dimaksudkan untuk memenuhi salah satu syarat kelulusan pendidikan strata 1 di Universitas Indonesia Fakultas Teknik Jurusan Teknik Elektro. Sedangkan tujuan dari penyusunan tugas akhir ini antara lain :

1. Menunjukkan bagaimana *routing protocol* berperan penting dalam memperkuat manajemen lalu lintas komunikasi data yang mendukung kemampuan *QoS* sebuah jaringan *VPN*.
2. Mengetahui pengaruh pembentukan/pemilihan rute oleh *routing* protokol *RIPv2, EIGRP* dan *OSPF* terhadap implementasi *IPSec* pada jaringan *MPLS*, baik pada jaringan stabil maupun tidak stabil dengan *bandwidth* yang minimum.
3. Mengetahui parameter *QoS* yang sangat berperan dalam perubahan performansi *video conference* yang dijalankan pada *routing protocol* yang berbeda, sebelum dan sesudah diimplementasikan *IPSec*.
4. Mengetahui *routing protocol* yang paling optimal dan paling efektif untuk jaringan *VPN MPLS* dengan tunnel *IPSec* berskala kecil, dilihat dari parameter jaringan yang terukur dan *MOS* dari responden.

1.4 Batasan Masalah

Dalam tugas akhir ini terdapat beberapa batasan masalah, yaitu :

1. Pengambilan data dilakukan pada jaringan berskala kecil (*testbed*), menggunakan 5 buah router dan 1 buah switch yang dibangun dengan skenario *"Site to site VPN Business"*.
2. Jaringan yang diamankan adalah koneksi antara *end-to-end* router, yaitu dari router *Head Office* sampai dengan *Branch Office*, sedangkan jaringan lokal di *Head Office* dan *Branch Office* diasumsikan aman.
3. Setiap *routing* protokol yang diimplementasikan, yaitu *RIPv2, EIGRP* dan *OSPF* digunakan parameter-parameter default.
4. *Bandwidth* yang diimplementasikan adalah *bandwidth* minimum untuk melakukan *video conference* disesuaikan dengan *codec* yang digunakan, yaitu 256 Kbps.

5. *Codec* yang digunakan untuk melakukan video conference adalah G.711 untuk audio dan H.263 untuk video.
6. Paket yang diukur dan dianalisa adalah paket *RTP* yang terdiri atas *codec* untuk audio G.711 dan video H.263, sedangkan paket lain yang tertangkap bersama paket *RTP* dibuang dan tidak dimasukkan dalam perhitungan analisa.
7. Data diambil pada dua kondisi yang berbeda, yaitu kondisi jaringan stabil dan jaringan tidak stabil. Jaringan stabil diperoleh pada saat melakukan video call dengan 2 user dan jaringan tidak stabil diperoleh pada saat melakukan *video conference* dengan 3 user.
8. Analisa performansi tidak menentukan *MOS*. Untuk *MOS* digunakan referensi dari responden.
9. Menggunakan IPV4 sebagai pengalamatannya.

1.5 Metodologi

1. Studi literatur

   Mengumpulkan dan mempelajari referensi tentang konsep jaringan, meliputi : *VPN*, *MPLS*, *IPSec*, algoritma *routing protocol RIP, EIGRP* dan *OSPF*.
2. Perancangan sistem

   Merancang sistem jaringan *VPN MPLS* dengan tunnel *IPSec*.
3. Implementasi sistem

   Implementasi dilakukan dengan 5 buah router *(testbed)* yang dibangun dengan skenario "*Site to site VPN Business*".
4. Pengambilan dan analisa data

   Setelah dilakukan implementasi, akan dilakukan pengujian terlebih dahulu menggunakan wireshark, kemudian dianalisa setiap parameter *QoS* untuk audio dan video-nya.
5. Penarikan kesimpulan

   Dari hasil analisa tersebut akan ditarik kesimpulan mengenai *routing protocol* mana yang paling optimal untuk diterapkan pada jaringan *VPN MPLS* dengan tunnel *IPSec* pada sistem jaringan yang telah dibuat.

6. Penulisan buku laporan

   Dalam penulisan laporan ini mengacu pada pedoman penulisan ilmiah, dalam hal ini penulisan tugas akhir yang bentuk bakunya telah diatur oleh pihak Universitas Indonesia.

1.6 Sistematika Penulisan

Sistematika penulisan tugas akhir ini adalah sebagai berikut :

BAB 1 PENDAHULUAN

Memuat penjelasan singkat mengenai latar belakang, perumusan masalah, tujuan penulisan, batasan masalah, metodologi, dan sistematika penulisan.

BAB 2 *VPN MPLS, IPSEC* DAN *DINAMIC ROUTING PROTOCOL*

Memuat berbagai materi yang diperlukan untuk mendasari pemahaman pada bagian-bagian selanjutnya dan yang akan digunakan untuk mencapai tujuan tugas akhir ini.

BAB 3 PERANCANGAN DAN IMPLEMENTASI JARINGAN *VPN* DENGAN *ROUTING PROTOCOL RIPv2, EIGRP* dan *OSPF*

Memuat rancangan keseluruhan sistem video call melalui jaringan *VPN MPLS* dengan tunnel *IPSec* dan implementasi *routing* protokol RIPv2, EIGRP dan OSPF yang dibagi dalam 3 skenario.

BAB 4 PENGUJIAN DAN ANALISA SISTEM

Memuat laporan realisasi dan penerapan rancangan yang sebelumnya telah dilakukan. sekaligus mengevaluasi hasil yang diperoleh.

BAB 5 PENUTUP

Memuat kesimpulan dan kemungkinan pengembangan untuk penelitian selanjutnya.

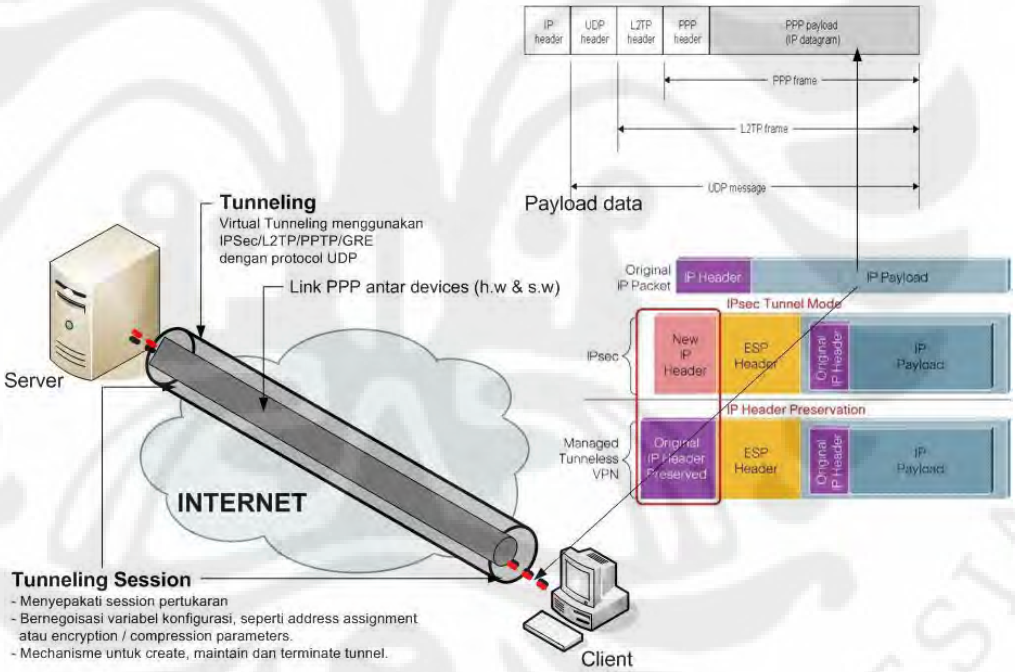
BAB 2

VPN MPLS, IPSEC DAN *DINAMIC ROUTING PROTOCOL*

2.1. *VPN*

Virtual Private Network (*VPN*) merupakan sebuah jaringan *private* yang menghubungkan satu *node* jaringan ke *node* jaringan lainnya dengan menggunakan jaringan publik (internet). Data yang dilewatkan akan dibungkus (*encapsulation*) dan dienkripsi agar terjamin kerahasiaannya.

Jaringan *VPN* dikoneksikan oleh penyedia jasa komunikasi (*Service Provider*) melalui routernya ke router-router lain dengan menggunakan jalur internet yang telah dienkripsi diantara dua titik, seperti yang ditunjukkan pada Gambar 2.1.



Gambar 2.1 Skema *tunneling* dan e*ncapsulation VPN* [6]

Sistem keamanan di *VPN* menggunakan beberapa lapisan, yaitu :

a) Metode *tunneling* (terowongan)

Membuat terowongan *virtual* di atas jaringan publik menggunakan protokol seperti *Point to Point Protocol* (*PPTP*), *Layer 2 Tunneling Protocol* (*L2TP*), *Generic Routing Encapsulation (GRE)* atau *IPSec*. *PPTP* dan *L2TP* adalah

Layer 2 *Tunneling Protocol*, keduanya melakukan pembungkusan *payload* pada *frame PPTP* untuk dilewatkan pada jaringan. Sedangkan *IPSec* berada di layer 3, menggunakan paket, dan akan melakukan pembungkusan *IP header* sebelum dikirim ke jaringan.

b) Metode enkripsi

Untuk membungkusn paket data yang lewat di dalam *tunneling*, data yang dilewatkan pada pembungkusan tersebut akan dirubah dengan algoritma kriptografi tertentu seperti *DES*, 3*DES* dan *AES*.

c) Metode autentikasi user

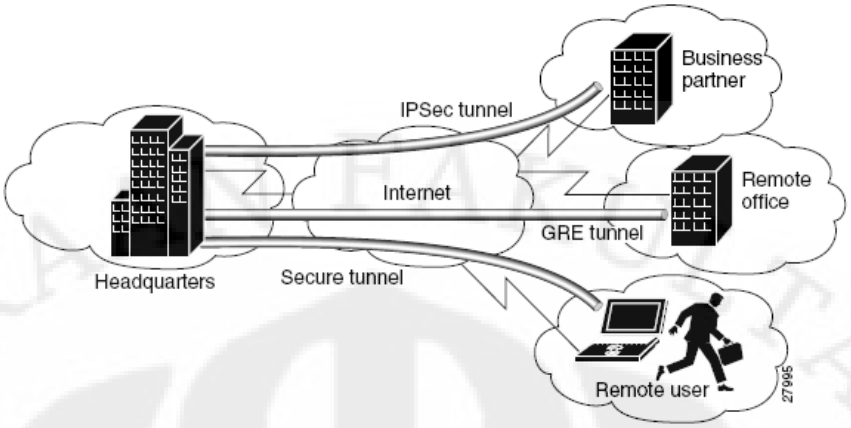
Karena banyak user yang akan mengakses biasanya digunakan beberapa metode autentikasi user seperti *Remote Acces Dial in user Services* (*RADIUS*) dan *Digital Certificates*.

d) Integritas data

Paket data yang dilewatkan pada jaringan publik perlu penjaminan integritas (keutuhan) data, apakah terjadi perubahan atau tidak. Metode *VPN* menggunakan *HMA C-MD5* atau *HMA C-SHA1* agar paket data tidak berubah pada saat pengiriman.

Ada 3 jenis implementasi jaringan *VPN* seperti yang ditunjukkan oleh Gambar 2.2, antara lain :

- *Remote access VPN* - menyediakan *remote access* ke jaringan *intranet* atau *extranet* perusahaan yang memiliki kebijakan yang sama sebagai jaringan privat. *Remote access VPN* memungkinkan user untuk dapat mengakses data perusahaan kapanpun dan di manapun mereka inginkan.
- *Intranet VPN* – menghubungkan antara kantor pusat suatu perusahaan dengan kantor cabang, kantor pembantu melalui *shared network* menggunakan koneksi yang permanen (*dedicated*).
- *Extranet VPN* – menghubungkan konsumen, *suppliers*, mitra bisnis, dan beberapa komunitas dengan kepentingan yang sama ke jaringan *intranet* perusahaan melalui infrastruktur yang terbagi menggunakan koneksi *dedicated*.



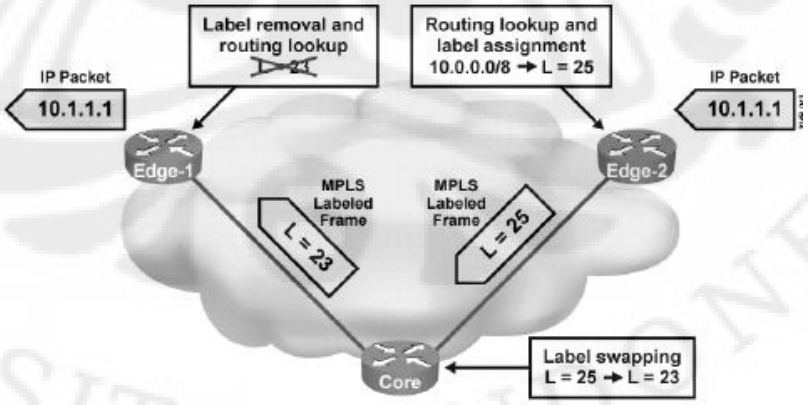
Gambar 2.2 *Extranet VPN, Intranet VPN* dan *Remote access VPN* [2]

2.2.*MPLS*

Multiprotocol Label Switching (*MPLS*) adalah teknologi penyampaian paket pada jaringan *backbone* (jaringan utama) berkecepatan tinggi yang menggabungkan beberapa kelebihan dari sistem komunikasi *circuit-switched* dan *packet-switched* yang melahirkan teknologi yang lebih baik dari keduanya.

MPLS adalah arsitektur jaringan yang didefinisikan oleh *IETF* untuk memadukan mekanisme *label swapping* di *layer* 2 dengan *routing* di *layer* 3 untuk mempercepat pengiriman paket. Paket-paket pada *MPLS* diteruskan dengan protokol *routing* seperti *OSPF*, *BGP* atau *EGP*, Protokol *routing* pada *layer* 3 sistem *OSI*, sedangkan *MPLS* berada diantara *layer* 2 dan 3.

Contoh konsep dasar *MPLS* ditunjukkan oleh Gambar 2.3.

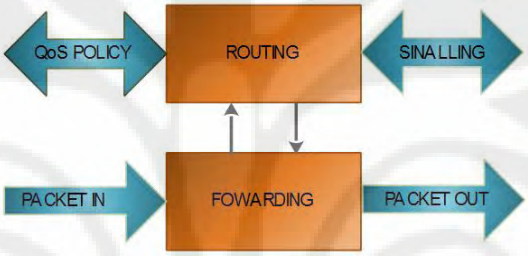


Gambar 2.3 Contoh konsep dasar *MPLS* [8]

2.2.1 Arsitektur *MPLS*

MPLS adalah arsitektur *network* yang didefinisikan oleh *IETF* untuk memadukan mekanisme *label swapping* di *layer* 2 dengan *routing* di *layer* 3 untuk mempercepat pengiriman paket. Jaringan *MPLS* terdiri atas sirkit yang disebut *Label-Switched Path (LSP),* yang menghubungkan titik-titik yang disebut *Label-Switched Router (LSR).*

Setiap *LSP* dikaitkan dengan sebuah *Forwarding Equivalence Class (FEC),* yang merupakan kumpulan paket yang menerima perlakuan *forwarding* yang sama di sebuah *LSR*. *FEC* didefinisikan dengan pemasangan label. Arsitektur *MPLS* dipaparkan dalam *RFC*-3031.



Gambar 2.4 Arsitektur *MPLS* dalam *RFC*-3031 [7]

Untuk membentuk *LSP*, diperlukan suatu protokol persinyalan. Protokol ini menentukan *forwarding* berdasarkan label pada paket. Label yang pendek dan berukuran tetap mempercepat proses *forwarding* dan mempertinggi fleksibilitas pemilihan jalur. Hasilnya adalah *network datagram* yang bersifat lebih *connection-oriented*.

2.2.2 Struktur Jaringan *MPLS*

Struktur jaringan *MPLS* terdiri dari *Edge Label Switching Routers* atau *Edge LSRs* yang mengelilingi sebuah *core Label Switching Routers (LSRs).* Adapun elemen-elemen dasar penyusun jaringan *MPLS* adalah :

- *Edge Label Switching Routers (ELSR)*

  ELSR ini terletak pada perbatasan jaringan *MPLS*, berfungsi untuk mengaplikasikan label ke dalam paket-paket yang masuk ke dalam jaringan *MPLS*. Sebuah *MPLS Edge Router* akan meganalisa *header IP*, dan akan menentukan label yang tepat untuk dienkapsulasi ke dalam paket tersebut ketika sebuah paket *IP* masuk ke dalam jaringan *MPLS*. Dan ketika paket yang

berlabel meninggalkan jaringan *MPLS*, maka *edge* router yang lain akan menghilangkan label tersebut (*label switches*). Perangkat *label switches* ini berfungsi untuk men-*switch* paket-paket yang telah dilabeli berdasarkan label tersebut. *Label switches* ini juga mendukung *layer* 3 *routing* ataupun *layer* 2 *switching* untuk ditambahkan dalam *label switching*. Operasi dalam *label switches* memiliki persamaan dengan teknik *switching* yang biasa dikerjakan dalam *ATM*.

- *Label Distribution Protocol (LDP)*

  LDP merupakan suatu prosedur yang digunakan untuk menginformasikan ikatan label yang telah dibuat dari sau *LSR* ke *LSR* lainnya dalam satu jaringan *MPLS*. Dalam arsitektur jaringan *MPLS*, sebuah *LSR* yang merupakan tujuan atau *hop* selanjutnya akan mengirimkan informasi tentang ikatan sebuah label ke *LSR* yang sebelumnya mengirimkan pesan untuk mengikat label tersebut bagi rute paketnya. Teknik ini biasa disebut distribusi label *downstream on demand*.

Jaringan ini memiliki beberapa keuntungan, diantaranya :

- *MPLS* mengurangi banyaknya proses pengolahan yang terjadi di *IP* routers, serta memperbaiki kinerja pengiriman suatu paket data.
- *MPLS* juga bisa mnyediakan *Quality of Service (QoS)* dalam jaringan *backbone*, dan menghitung parameter *QoS* menggunakan teknik *Differentiated services (Diffserv)* sehingga setiap layanan paket yang dikirimkan akan mendapat perlakuan yang berbeda sesuai dengan skala prioritasnya.

2.3. *IPSec Tunnel*

2.3.1. Arsitektur *IPSec*

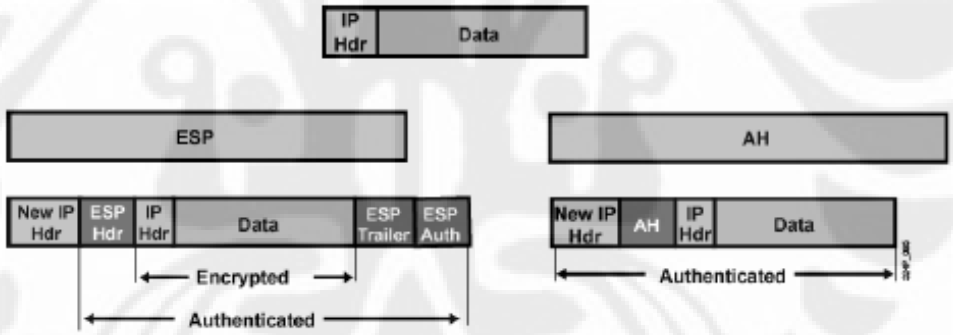
IPSec merupakan jenis protokol yang mengintegrasikan fitur *security* yang meliputi proses *authentifikasi*, integritas, dan kepastian ke dalam *IP* (*Internet Protocol*). Dimana proses tersebut dilakukan pada *network layer* atau *layer* ketiga dalam model OSI.

IPSec Tunnel memperbolehkan *peer* untuk mengirimkan informasi secara aman melalui jaringan *IP public* atau jaringan yang tidak dipercayai. Dengan menggunakan *IPSec Tunnel*, kita dapat melakukan enkripsi dan atau membuat

media komunikasi (*tunnel*) terautentifikasi tergantung pada kondisi protokol yang diinginkan oleh dua *peer* tersebut. Protokol tersebut antara lain :

- *Authentication Header (AH),* autentifikasi sumber data dan proteksi terhadap pencurian data. Protokol *AH* dibuat dengan melakukan enkapsulasi paket *IP* asli ke dalam paket baru yang mengandung *IP Header* yang baru yaitu *AH Header* disertai dengan *header* yang asli. Isi data yang dikirimkan melalui protokol *AH* bersifat *clear text* sehingga *tunnel* yang berdasar pada protokol *AH* ini tidak menyediakan kepastian data [*RFC*-2402].
- *Encapsulated Security Payload (ESP)*, dapat menyediakan kepastian data, autentifikasi sumber data dan proteksi terhadap gangguan pada data. Protokol *ESP* dibuat dengan melakukan enkripsi pada paket *IP* dan membuat paket *IP* lain yang mengandung *header IP* asli dan *header ESP*. Data yang terenkripsi (yang mengandung *header IP* asli) dan *trailer ESP*, separuhnya terenkripsi dan sebagian tidak [*RFC*-2406].

  Gambar 2.5 menunjukkan perbedaan *header ESP* dan *AH*.



Gambar 2.5 *Header ESP* dan *AH* [8]

Salah satu contoh penggunaan *IPSec Tunnel* ialah pada *VPN*. Dimana *VPN* memperbolehkan komunikasi data yang aman melalui jaringan publik atau jaringan yang tidak dipercayai.

Spesifikasi *IPSec* memperbolehkan kedua protokol *AH* dan *ESP* untuk diaplikasikan pada data yang ingin dilindungi, tetapi dengan menggunakan *IPSec Tunnel* yang berdasar pada protokol *ESP*, hal tersebut tidak dibutuhkan. *ESP* menyediakan algoritma autentifikasi yang disebut dengan *"authenticator"* yang dapat digunakan sebagai autentifikasi sumber data. *authenticator* ini, dapat diset

null atau *ESP* tidak melakukan autentifikasi sumber data. Dalam hal ini dapat digunakan metode autentifikasi berdasarkan protokol *AH*.

Protokol *AH* dan *ESP* berdasar pada beberapa kunci yang terenkripsi dan algoritma hanya untuk menyediakan integritas, autentifikasi, dan kepastian. Spesifikasi *IPSec* tidak hanya berdasar pada algoritma mana yang harus dipergunakan, tapi dapat pula disertakan beberapa algoritma lain yang didukung oleh O*perating System* di mana *IPSec* ini akan diimplementasikan. Beberapa algoritma yang sering dipakai untuk kedua protokol tersebut antara lain meliputi *HMAC-MD5* dan *HMAC-SHA*.

Tunnel yang berdasar pada protokol *ESP* menggunakan algoritma *Data Encryption Standard* (*DES*) atau *Tipple Data Encryption Standard* (3*DES*).

2.3.2 *Security Association*

Kombinasi tentang bagaimana melindungi data (*ESP* dan atau *AH* termasuk algoritma dan kunci), apa data yang dilindungi dan pada saat apa data dilindungi disebut dengan *Security Association (SA).*

SA merupakan identifikasi unik berbasiskan *Security Parameter Index (SPI),* alamat tujuan IP dan protokol keamanan (*AH* dan atau *ESP*) yang diimplementasikan dalam trafik jaringan *IPSec*. Dua tipe *SA* yang didefinisikan yaitu :

- *Transport Mode*

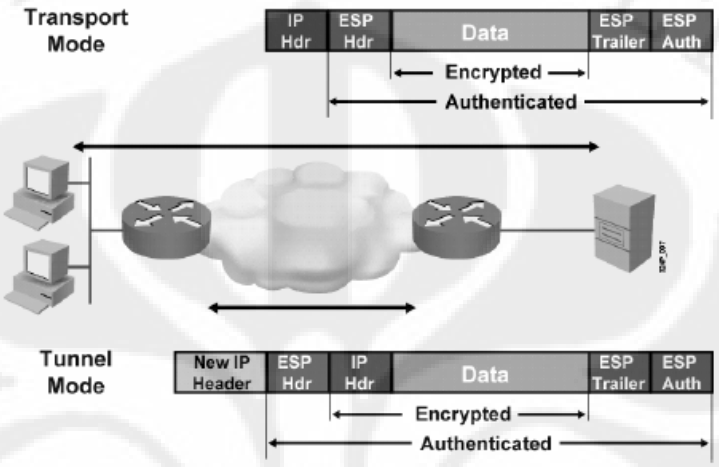
  Pada mode ini, *SA* diimplementasikan pada trafik antara dua *host*. Pada kasus *ESP, transport mode SA* memberikan pelayanan keamanan hanya pada *layer* tersebut tidak untuk *IP Header* atau *header* lain yang tidak mendahului *header ESP*. Pada kasus *AH*, perlindungan juga diberikan pada *IP Header* atau *header* tambahan lainnya.

- *Tunnel Mode*

  Merupakan *SA* yang diimplementasikan pada dua *gateway IPSec Tunnel*. Pada mode ini, terdapat *IP Header* tambahan di luar yang menspesifikasikan tujuan pemrosesan *IPSec* ditambah *IP Header* tambahan di dalam yang menunjukkan alamat tujuan paket yang sebenarnya. *Header* protokol keamanan akan tampak pada bagian setelah *IP Header* tambahan di luar dan sebelum *IP Header* tambahan di dalam.

Pada kasus *AH*, bagian *IP Header* tambahan di luar diberikan perlindungan seperti paket *IP* yang di *tunnel*. Pada kasus *ESP*, proteksi hanya diberikan pada paket yang di-*tunnel* saja.

Contoh konsep *IPSec* pada mode *transport* dan mode *tunnel* ditunjukkan oleh Gambar 2.6.



Gambar 2.6 *ESP* dalam mode *transport* dan mode *tunnel* [8]

2.3.3 Penentuan Algoritma *Internet Key Exchange (IKE)*

IKE tunnel melakukan negoisasi kunci yang akan dipergunakan diantara *peer* serta melakukan negoisasi dalam penentuan algoritma protokol (*AH* dan atau *ESP*) yang akan dipakai. Selain itu, *IKE tunnel* menggunakan algoritma Diffie-Hellman untuk menciptakan kunci yang simetris antar *peer* dalam membentuk *tunnel*. Kunci ini hanya akan berlaku sepanjang nilai *time to live*, setelah itu kunci baru akan dinegoisasikan kembali. *Tunnel* yang terbentuk disebut *IKE Tunnel* atau *Tunnel* Negoisasi.

Proses untuk menentukan algoritma yang digunakan pada *IKE* meliputi beberapa hal yaitu :

- Integritas

  Integritas data didukung dengan penentuan *hash algorithm* yang akan dipakai pada proses negosiasi, meliputi :

  - ✓ *SHA (Secure Hash Standard)*

    SHA memproduksi 160 bit *digest*, di mana hasil *hash function* tersebut akan lebih tahan terhadap proses *brute force* daripada

MD5, tetapi proses ini membutuhkan *resource* yang lebih banyak daripada MD5.

- ✓ MD5

  MD5 memproduksi 128 bit *digest* di mana waktu pemrosesan lebih cepat dibandingkan performansi *SHA* tetapi lebih lemah dibandingkan *SHA*. Salah satu operasi hash MD5 : MD5 yang terdiri dari 64 bit operasi ini, dikelompokkan pada 4 ronde masing-masing 16 operasi.

- Autentifikasi

  Proses autentifikasi dapat dilakukan dengan cara sebagai berikut :

  - ✓ *Certificate (RSA Encryption)*

    Konfigurasi ini akan memperbolehkan dua *peer* untuk melakukan autentifikasi dengan berbagi kunci public. Konfigurasi ini dapat melakukan penyangkalan pada negoisasi *IKE*. *RSA Encrytion* dibutuhkan untuk melakukan pemeriksaan *authority* saja.

  - ✓ *Certificate (RSA Signature)*

    Konfigurasi ini dapat melakukan penyangkalan juga sehingga pihak ketiga harus dibuktikan dahulu dengan *RSA Signature* ini. Tingkat keamanan konfigurasi ini di bawah *Certificate* dengan *RSA Encryption.*

  - ✓ *Shared Secret*

    Konfigurasi ini akan membutuhkan *pre-shared-key* daripada *certificate*. Konfigurasi ini lebih mudah namun dalam jaringan yang besar waktu yang diperlukan lebih besar.

- Kepastian

  - ✓ *DES*

    DES lebih cepat dibandingkan *triple DES* dan membutuhkan *resource* yang lebih sedikit tetapi kurang aman. Jika kita membutuhkan kepastian data dengan memperhatikan faktor *resource* dan kecepatan maka pilihan ini lebih baik.

- ✓ Triple *DES (3DES)*

  Triple *DES* bukan merupakan bagian dari *IPSec* standar, karena implementasinya belum dilakukan secara menyeluruh. Kecepatan yang dibutuhkan lebih lambat dan membutuhkan *resource* lebih banyak untuk melakukan tiga kali penghitungan.

- Penurunan Kunci

  Yaitu pembentukan awal kunci berdasarkan *Group Diffie-Hellman (dh-group)*

  - ✓ *Group* 1

    Menggunakan modulus 768 bit (mod768)
  - ✓ *Group* 2

    Menggunakan modulus 1024 bit (mod1024)
  - ✓ *Group* 5

    Menggunakan modulus 1536 bit (mod1536)
  - ✓ *Group* 14

    Menggunakan modulus 2048 bit (mod2048)
  - ✓ *Group* 15

    Menggunakan modulus 3072 bit (mod3072)
  - ✓ *Group* 16

    Menggunakan modulus 4096 bit (mod4096)
  - ✓ *Group* 17

    Menggunakan modulus 6144 bit (mod6144)
  - ✓ *Group* 18

    Menggunakan modulus 8192 bit (mod8192)

2.4 Routing

Dalam suatu sistem *packet switching*, *routing* mengacu pada proses pemilihan jalur untuk pengiriman paket, dan router adalah perangkat yang melakukan tugas tersebut. Perutean dalam *IP* melibatkan baik gateway maupun host yang ada. Ketika suatu program aplikasi dalam suatu *host* akan berkomunikasi, protokol *TCP/IP* akan membangkitkannya dalam bentuk banyak

datagram. *Host* harus membuat keputusan perutean untuk memilih jalur pengiriman.

2.4.1 Protokol *Routing*

Protokol *routing* adalah protokol yang digunakan oleh router-router untuk saling bertukar informasi *routing*. Router-router pada jaringan *TCP/IP* membentuk tabel *routing* berdasarkan informasi *routing* yang dipertukarkan setiap selang waktu tertentu. Protokol *routing* mempunyai kemampuan untuk membangun informasi dalam tabel *routing* secara dinamik. Apabila terjadi perubahan jaringan protokol *routing* mampu memperbaharui informasi *routing* tersebut. *Routing* pada jaringan *TCP/IP* dibagi menjadi 2 macam :

- *Interior Gateway Protocol (IGP)*

  Merupakan protokol *routing* yang menangani perutean dalam suatu *autonomous system.*
- *Exterior Gateway Protocol (EGP)*

  Merupakan protokol *routing* yang menangani *routing* antar *autonomous system. Autonomous system* adalah suatu sistem jaringan internet yang berada dalam satu kendali administrasi dan teknis.

Ada beberapa macam protokol *routing* yang sering digunakan, diantaranya :

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

Berdasarkan pembagian utamanya maka protokol *routing* yang termasuk dalam *IGP* adalah *RIP*, *EIGRP*, dan *OSPF*, sedangkan yang termasuk dalam *EGP* adalah *BGP*. *EGP* memiliki kemampuan untuk menentukan *policy routing* karena sebagian *autonomous system* di internet mempunyai kebijakan dalam hal *routing*. Untuk pelaksanaan *routing* dalam *IGP policy* ini tidak diperlukan.

2.4.2 Klasifikasi Algoritma *Routing*

Algoritma *routing* digunakan untuk membangun dan mengatur tabel *routing*. Tipe informasi yang ada pada tabel *routing* antara lain :

- *Direct route* yang diperoleh dari *interface* yang terpasang.

- *Indirect route* yang dicapai melalui sebuah atau beberapa *gateway.*
- *Default route*, yang merupakan arah akhir apabila tidak bisa terhubung melalui *direct* maupun *indirect route.*

Terdapat 2 cara untuk membangun tabel *routing*, yaitu :

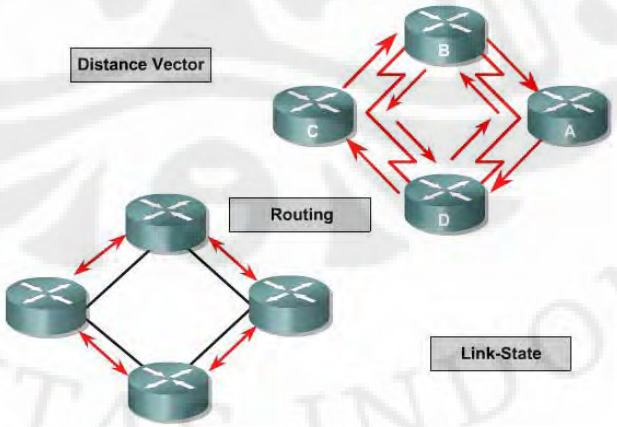
- *Static Routing*

  Routing ini dibangun berdasarkan definisi dari administrator. Pengisian dan pemeliharaan tabel *routing* dilakukan secara manual oleh administrator.
- *Dinamic Routing*

  Routing ini dibangun berdasarkan informasi yang dikumpulkan oleh protokol *routing*. Protokol ini didesain untuk mendistribusikan informasi yang secara dinamis mengikuti perubahan kondisi jaringan. Router saling bertukar informasi *routing* agar dapat mengetahui alamat tujuan dan menerima tabel *routing*. Pemeliharaan jalur dilakukan berdasarkan pada jarak terpendek antara perangkat tujuan dan perangkat tujuan.

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

- *Distance vector*, bertujuan untuk menentukan arah atau vector dan jarak ke link-link lain dalam suatu *internetwork.*
- *Link State*, bertujuan untuk menciptakan kembali topologi yang benar pada suatu *internetwork*.



Gambar 2.7 Klasifikasi algoritma *routing* [12]

2.4.3 *RIPv2*

Routing Information Protocol (RIP) merupakan salah satu protokol *routing distance vector*, sebuah protokol yang sangat sederhana. Dikarenakan RIP berdasarkan *open standard* dan mudah diimplementasikan. Protokol *distance vector* sering juga disebut protokol Bellman-Ford, karena berasal dari algoritma perhitungan jarak terpendek oleh R.E Bellman, dan didistribusikan dalam bentuk algoritma terdistribusi pertama kali oleh Ford dan Fulkerson. *RIP* didefinisikan pada *RFC*-1058 dimana karakteristik dari *RIP* ini adalah sebagai berikut :

- Termasuk jenis *Distance Vector Protocol*
- *Classfull routing protocol*
- Menggunakan Hop untuk penentuan jalur terbaik ke jaringan tujuan
- Informasi *routing* dikirimkan secara regular setiap 30 detik

RIP versi 2 atau yang lebih dikenal dengan *RIPv2* seperti tertuang didalam *RFC*-1723 lahir dari fakta kekurangan-kekurangan yang terdapat pada *RIP* versi terdahulu. *RIPv2* ini bukan merupakan sebuah protokol *routing* baru melainkan merupakan pengembangan dari versi sebelumnya. Dasar yang digunakan sama seperti *RIP* terdahulu hanya saja terdapat beberapa tambahan didalam *RIPv2*.

2.4.3.1 Kompatibilitas *RIPv2* dengan *RIPv1*

Tabel berikut ini menggambarkan perbandingan antara *RIPv1* (versi terdahulu) dengan *RIPv2*.

Tabel 2.1 Perbandingan antara *RIPv1* dan *RIPv2* [14]

	RIPv1	RIPv2
Protokol Routing	Classful	Classless
Mendukung VLSM	Tidak	Ya
Membawa informasi Subnet Mask dalam *routing* informasi	Tidak	Ya
Mendukung Autentikasi	Tidak	Ya
Mendukung Manual Route Summarization	Tidak	Ya
Tipe addressing	Broadcast	Multicast

Pengembangan utama pada *RIPv2* adalah sebagai berikut :

- Autentikasi

RIPv2 mendukung dilakukannya autentikasi. Kebutuhan akan autentikasi didalam jaringan sangat penting karena hal ini mencegah seseorang yang tidak berhak melakukan akses kedalam jaringan dan kemudian memberikan informasi *routing* palsu.

- *Subnet Mask*

  Subnet mask inilah yang menjadi ide awal dari pengembangan *RIP* versi 1 ke *RIP* versi 2. Dengan kemampuan membawa subnet mask kedalam informasi *routing* membuat *RIP* versi 2 mampu untuk mendukung berbagai variabel subnet mask pada jaringan (*VLSM*) dan tidak terbatas hanya kepada tipe utama dari pembagian kelas *ip address* yang artinya juga mendukung *Classless Inter Domain Routing (CIDR).*

- *Next-hop address*

  Dengan adanya pengembangan ini *RIP* versi 2 menjadi lebih efisien didalam pemilihan hop selanjutnya untuk meneruskan data mencapai tujuannya dan memungkinkan untuk optimalisasi rute dalam sebuah jaringan dimana jaringan tersebut menggunakan berbagai macam protokol *routing*.

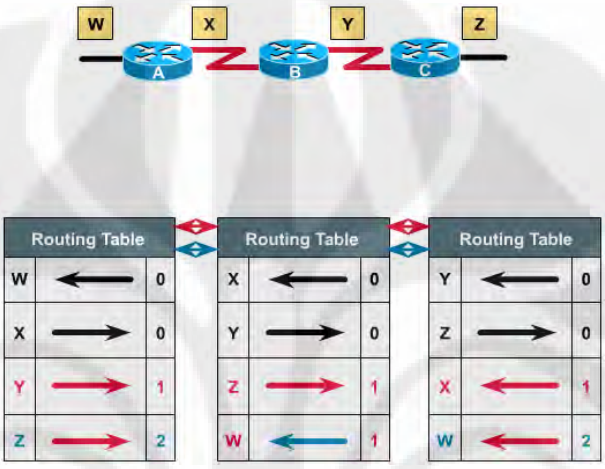
- *Multicast*

  Jika *RIP* versi terdahulu menggunakan *broadcast,* maka *RIP* versi 2 menggunakan *multicast* didalam menyebarkan informasi *routing*. Dengan penggunaan *multicast* didalam jaringan dapat mengurangi beban pada saat menyebarkan informasi *routing* karena hanya perangkat-perangkat yang menjalankan *routing* serupa saja yang akan menerima informasi *routing* tersebut.

2.4.3.2 Cara Kerja *RIP*

RIP mengirim pesan *routing-update* pada *interval* yang tetap. Ketika router menerima *routing-update* yang berisi perubahan *routing*, ia mengupdate tabel *routing*-nya ke rute yang baru. Dalam hal ini *metric* yang diterima bertambah nilainya 1, dan *interface* asal dari update menunjukkan hop berikutnya dalam tabel *routing*. Router-router *RIP* memperbaiki tidak hanya rute yang terbaik saja ke tujuan tapi juga memperbaiki rute ke tujuan yang nilainya sama. *RIP* merupakan *time-driven*, tapi di implementasi Cisco *RIP* mengirim *triggered*

update kapanpun kalau perubahan dideteksi. Topologi mengalami perubahan juga akan dikirim *triggered update*-nya langsung. Tanpa *trigger*, *RIP* tidak akan bagus unjuk kerjanya. Setelah proses update tabel *routing* terjadi, maka konfigurasi pun mengalami perubahan, kemudian router secara langsung memulai transmit update *routing* untuk menginformasikan ke router-router lainnya tentang perubahan yang terjadi. *Triggered update* ini, dikirim secara regular dan terjadwal.



Gambar 2.8 Algoritma *RIPv2* [12]

2.4.4 *OSPF*

Open Shortest Path First (*OSPF*) adalah salah satu *routing protocol link-state* yang dikembangkan sebagai pengganti *distance vector routing protocol RIP*. *RIP* adalah *routing protocol* yang cocok pada awal perkembangan jaringan dan internet. Tetapi ini tergantung pada *hop count* sebagai pengukuran dalam memilih rute terbaik dan tercepat, tapi kemudian ini tidak sesuai lagi seiring dengan bertambah luasnya jaringan yang memerlukan solusi *routing* yang sangat cepat. *OSPF* adalah *classless routing protocol* yang menggunakan konsep area untuk skalabilitas. *RFC*-2328 mendefinisikan *OSPF metric* sebagai nilai penentu yang biasa dikenal sebagai *cost*. CISCO *IOS* menggunakan *bandwidth* sebagai *OSPF cost metric*.

Keuntungan utama *OSPF* dibandingkan *RIP* adalah *OSPF* dapat melakukan konvergensi yang cepat dan skalabilitas lebih luas untuk implementasi jaringan yang lebih besar.

2.4.4.1 Tipe-tipe Paket *OSPF*

Tipe Paket *OSPF* dapat didefinisikan sebagai berikut :

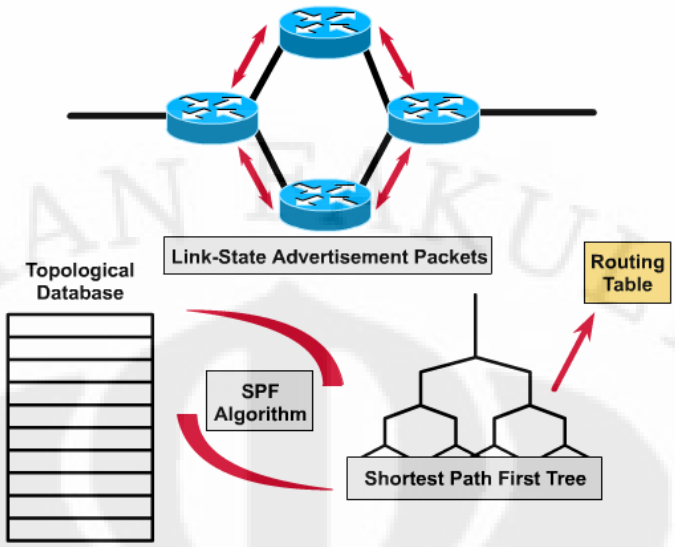
1. *Hello* – Paket *Hello* digunakan untuk membangun dan memelihara *adjancency* dengan router *OSPF* lainnya.
2. *DBD – Database Description (DBD)* berisi daftar-daftar dari database *link state* router pengirim dan digunakan oleh router penerima untuk memeriksa dan dibandingkan dengan database *link state local.*
3. *LSR – Receiving Routers* kemudian bisa meminta informasi lebih lanjut tentang isi di dalam *DBD* dengan mengirim *Link-State Request (LSR).*
4. *LSU – Link State Update (LSU)* paket digunakan untuk me-*reply* ke *LSRs* serta mengumumkan informasi baru. *LSUs* berisi tujuh jenis *Link-State Advertisements (LSAs)* yang berbeda.
5. *LSAck* – Ketika sebuah *LSU* diterima, router mengirim sebuah *Link-State Acknowledgement (LSAck)* sebagai konfirmasi penerimaan *LSU.*

2.4.4.2 Pemilihan *DR* dan *BDR*

Untuk mengurangi jumlah lalu lintas di *multi-acces* jaringan *OSPF*, *OSPF* memilih sebuah *Designated Router (DR)* dan *Backup Designated Router (BDR)*. *DR* bertangung jawab untuk memperbaharui semua router *OSPF* yang lain (disebut *DRothers)* ketika terjadi perubahan pada jaringan *multi-access*. *BDR* akan memonitor *DR* dan mengambil alih sebagai *DR* jika terjadi kegagalan pada *DR.* Dalam gambar, R1,R2, dan R3 dihubungkan melalui titik *point-to-point* link .Oleh karena itu, tidak terjadi pemilihan *DR/BDR.*

2.4.4.3 Algoritma *OSPF*

Setiap router *OSPF* menjaga sebuah *link-state database* berisi *Link-state advertisement* (*LSAs*) yang diterima dari semua router yang lain. Satu kali router menerima semua *LSAs* dan membuat *local link-state* databasenya, *OSPF* mengggunakan algoritma Dijkstra *Shortest Path First (SPF)* untuk membuat pohon *SPF.* Pohon *SPF* kemudian digunakan untuk membuat tabel *IP routing* yang berisi daftar jalan yang terbaik menuju setiap jaringan.



Gambar 2.9 Algoritma *OSPF* [12]

2.4.5 *EIGRP*

Enhanced Interior Gateway Protocol (EIGRP) mengkombinasikan kelebihan-kelebihan yang dimiliki oleh protokol *routing link-state* dan *distance vector*. Tetapi pada dasarnya *EIGRP* adalah protokol *distance vector* karena router-router yang menjalankan *EIGRP* tidak mengetahui *road map*/topologi jaringan secara menyeluruh seperti pada protokol *link-state*.

EIGRP mudah dikonfigurasi seperti pendahulunya *(IGRP)* dan dapat diadaptasikan dengan variasi *topology network*. Penambahan fitur-fitur protokol *link-state* seperti *neighbor discovery* membuat *EIGRP* menjadi protokol *distance vector* tingkat lanjut. *EIGRP* menggunakan algoritma *Diffusing Update Algoritm* (*DUAL*) sebagai mesin utama yang menjalankan linkungan *EIGRP*, *DUAL* dapat diperbandingkan dengan algoritma *SPF* Dijkstra pada *OSPF*.

2.4.5.1 Fitur-fitur *EIGRP*

EIGRP memiliki fitur-fitur utama sebagai berikut:

- *Partial updates* : *EIGRP* tidak mengirimkan update secara periodik seperti yang dilakukan oleh RIP, tetapi *EIGRP* mengirimkan update hanya jika terjadi perubahan *route/metric* (*triggered update*). *Update* yang dikirimkan hanya berisi informasi tentang route yang menggalami perubahan saja. Pengiriman pesan update ini juga hanya ditujukan sebatas pada router-router yang membutuhkan informasi perubahan tersebut saja. Hasilnya

EIGRP menghabiskan *bandwidth* yang lebih sedikit daripada *IGRP*. Hal ini juga membedakan *EIGRP* dengan protokol *link-state* yang mengirimkan update kepada semua router dalam satu area.

- *Multiple network – layer protocol support*: *EIGRP* mendukung protokol *IP*, AppleTalk, dan Novell Netware IPX dengan memanfaatkan modul-modul yang tidak bergantung pada protokol tertentu

Fitur *EIGRP* lain yang patut diperhatikan adalah sebagai berikut :

- Koneksi dengan semua jenis data link dan topologi tanpa memerlukan konfigurasi lebih lanjut, *routing protocol* lain seperti *OSPF*, menggunakan konfigurasi yang berbeda untuk protokol *layer* 2 (*Data Link*) yang berbeda, misalnya Ethernet dan Frame Relay. *EIGRP* beroperasi dengan efektif pada lingkungan *LAN* dan *WAN*. Dukungan *WAN* untuk link *point-to-point* dan topologi non *Broadcast MultiAaccess (NBMA*) merupakan standar *EIGRP*.
- *Metric* yang canggih : *EIGRP* menggunakan algoritma yang sama dengan *IGRP* untuk menghitung *metric* tetapi menggambarkan nilai-nilai dalam format 32-bit. *EIGRP* mendukung *load balancing* untuk *metric* yang tidak seimbang *(unequal),* yang memungkinkan *engineer* untuk mendistribusikan *traffic* dalam *network* dengan lebih baik.
- *Multicast* dan *unicast* : *EIGRP* menggunakan *multicast* dan *unicast* sebagai ganti *broadcast.* Address *multicast* yang digunakan adalah 224.0.0.10.

2.4.5.2 Proses dan Teknologi *EIGRP*

EIGRP menggunakan 4 teknologi kunci yang berkombinasi untuk membedakan *EIGRP* dengan protocol *routing* yang lainnya : *neighbor discovery/recovery*, *Reliable Transport Protocol (RTP), DUAL Finite State Machine,* dan *protocol-dependent modules*.

- *Neighbor discovery/recovery*

  Menggunakan paket *hello* antar *neighbor*
- *Reliable Transport Protocol (RTP)*

  Pengiriman paket yang terjamin dan terurut kepada semua *neighbor*

- *DUAL finite-state machine*

  Memilih jalur dengan *cost* paling rendah dan bebas *looping* untuk mencapai *destination*
- *Protocol-dependent module (PDM)*
  - *EIGRP* dapat mendukung IP, AppleTalk, dan Novell Netware.
  - Setiap protocol disediakan modul *EIGRP* tersendiri dan beroperasi tanpa saling mempengaruhi satu sama lain.

2.4.5.3 Cara Kerja *EIGRP*

Istilah- istilah algoritma *DUAL* antara lain :

- Memilih jalur/route untuk mencapai suatu *network* dengan *metric* paling rendah, dan bebas *looping.*
- *Advertised Distance* (*AD*), menggambarkan seberapa jauh sebuah *network* dari *neighbor*, merupakan *metric* antara router *next-hop* dengan *network destination.*
- *Feasible distance* (*FD*), menggambarkan seberapa jauh sebuah *network* dari router, merupakan ongkos (*metric*) antara router dengan router *next-hop* ditambah dengan *AD* dari router *next-hop*.
- Ongkos paling rendah = *FD* paling rendah.
- *Successor*, adalah jalur utama untuk mencapai suatu *network* (router terbaik), merupakan router *next-hop* dengan *next-hop* dengan ongkos paling rendah dan jalur bebas *looping.*
- *Feasible Successor*, adalah jalur backup dari *successor* (*AD* dari *feasible successor* harus lebih kecil daripada *FD* dari *successor*).

EIGRP menggunakan dan memelihara 3 jenis tabel. Tabel *neighbor* untuk mendaftar semua router *neighbor*, tabel topologi untuk mendaftar semua entri route untuk setiap network *destination* yang didapatkan dari setiap *neighbor*, dan tabel *routing* yang berisi jalur/route terbaik untuk mencapai ke setiap *destination.*

1) *Neighbor Table*

   Ketika router menemukan dan menjalin hubungan *adjency* (ketetanggaan) dengan *neighbor* baru, maka router akan menyimpan address router *neighbor* beserta *interface* yang dapat menghubungkan dengan *neighbor*

tersebut sebagai satu entri dalam tabel *neighbor*. Tabel *neighbor EIGRP* dapat diperbandingkan dengan *database adjacency* yang digunakan oleh protokol *routing link-state* yang keduanya mempunyai tujuan yang sama: untuk melakukan komunikasi 2 arah dengan setiap *neighbor* yang terhubung langsung.

Ketika *neighbor* mengirimkan paket *hello*, ia akan menyertakan informasi hold time, yakni total waktu sebuah router dianggap sebagai *neighbor* yang dapat dijangkau dan operasional . Jika paket *hello* tidak diterima sampai hold time berakhir, algoritma *DUAL* akan menginformasikan terjadinya perubahan topologi.

2) *Topology Table*

Ketika router menemukan *neighbor* , maka router akan mengirimkan sebuah update mengenai rute-rute yang ia ketahui yang ia ketahui kepada *neighbor* baru tersebut dan juga sebaliknya menerima informasi yang sama dari *neighbor*. Update-update inilah yang akan membangun tabel topologi. Tabel topologi berisi informasi semua *network destination* yang di-*advertise* oleh router *neighbor*. Jika *neighbor* meng-*advertise* route ke suatu *network destination*, maka *neighbor* tersebut harus menggunakan route tersebut untuk meneruskan paket.

Tabel topologi di update setiap kali ada perubahan pada *network* yang terhubung langsung atau pada *interface* atau ada pemberitahuan perubahan pada suatu jalur dari router *neighbor*.

Entri pada tabel topologi untuk suatu *destination* dapat bersatus aktif dan pasif. *Destination* akan berstatus pasif jika router tidak melakukan komputasi ulang, dan berstatus aktif jika router masih melakukan komputasi ulang. Jika selalu tersedia *feasible successor* maka *destination* tidak akan pernah berada pada status aktif dan terhindar dari komputasi ulang. Status yang diharapkan untuk setiap *network destination* adalah status pasif.

3) *Routing table*

Router akan membandingkan semua *FD* untuk mencapai *network* tertentu dan memilih jalur/route dengan *FD* paling rendah dan meletakkannya pada

tabel *routing*, jalur/route inilah yang disebut *successor route*. *FD* untuk jalur/route yang terpilih akan menjadi *metric EIGRP* untuk mencapai *network* tersebut dan disertakan dalam tabel *routing*.

2.4.5.4 Paket-paket *EIGRP*

EIGRP saling berkomunikasi dengan tetangga-nya secara *multicast* (224.0.0.10) dan menggunakan 5 jenis pesan (*message*) dalam berhubungan dengan *neighbor*-nya :

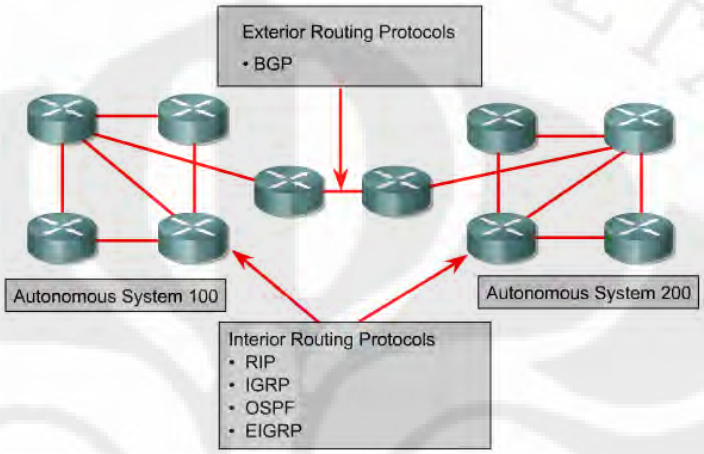
- *Hello* : Router-router menggunakan paket *Hello* untuk menjalin hubungan *neighbor*. Paket – paket dikirimkan secara *multicast* dan tidak membutuhkan.
- *Update* : Untuk mengirimkan update informasi *routing*. Tidak seperti RIP yang selalu mengirimkan keseluruhan tabel *routing* dalam pesan update, *EIGRP* menggunakan triggered update yang berarti hanya mengirimkan update jika terjadi perubahan pada *network* (misal: ada *network* yang *down*). Paket update berisi informasi perubahan jalur/rute. Update-update ini dapat berupa *unicast* untuk router tertentu atau *multicast* untuk beberapa router yang terhubung.
- *Query* : Untuk menanyakan suatu route kepada tetangga. Biasanya digunakan saat setelah terjadi kegagalan/*down* pada salah satu router *network*, dan tidak terdapat *feasible successor* untuk rute/jalur tersebut. Router akan mengirimkan pesan *query* untuk memperoleh informasi route alternatif untuk mencapai *network* tersebut, biasanya dalam bentuk *multicast* tapi bisa juga dalam bentuk *unicast* untuk beberapa kasus tertentu.
- Reply : Respon dari pesan *Query*.
- *ACK :* Untuk memberikan *acknowledgment* (pengakuan/konfirmasi) atas pesan update, *query,* dan *reply*.

2.4.6 *BGP*

Border Gateway Protocol (BGP) adalah merupakan *routing protocol exterior*. *BGP* digunakan untuk menghindari *routing loop* pada jaringan jaringan

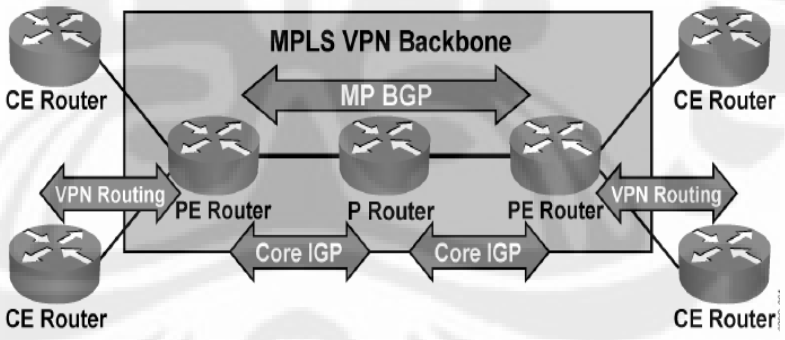
internet. Standar *BGP* menggunakan *RFC*-1771 yang berisi tetntang *BGP* versi 4. *BGP* mempunyai karakteristik sebagai berikut:

- Menggunakan *routing protocol distance vector*
- Digunakan antara *ISP* dengan *ISP* dan *client-client*
- Digunakan untuk merutekan trafik internet antar *autonomous system*



Gambar 2.10 Hubungan antara *BGP* dengan *routing protocol IGP* [12]

Analogi routing pada jaringan MPLS ditunjukkan pada Gambar 2.11 dibawah ini.



Gambar 2.11 *Routing* pada *VPN MPLS* [8]

2.5 Konsep dasar *Video Conference*

Dalam perencanaan untuk mengimplementasikan *video conference* pada *Local Area Network (LAN),* perlu memperhitungkan kebutuhan *bandwidth*, karena saat pengiriman video estimasi alokasi *bandwidth* menjadi sangat penting karena akan memakan sebagian besar *bandwidth* komunikasi yang ada. Sehingga teknik-

teknik untuk melakukan kompresi data menjadi sangat strategis untuk memungkinkan penghematan *bandwidth* komunikasi.

Sebagai gambaran sebuah kanal gambar (video) yang baik tanpa di kompresi akan mengambil *bandwidth* sekitar 9 Mbps, sedangkan sebuah kanal suara (audio) yang baik tanpa di kompresi akan mengambil *bandwidth* sekitar 64 Kbps. Dari gambaran di atas dapat diasumsikan bahwa kebutuhan minimal *bandwidth* yang dibutuhkan untuk mengirimkan gambar dan suara adalah sebesar 9.064 Mbps, memang akan membutuhkan *bandwidth* yang sangat lebar. Namun dengan teknik kompresi yang ada, sebuah kanal suara dan gambar sebelum dilewatkan dalam jaringan *TCP/IP* akan terlebih dahulu melalui proses kompresi sehingga dapat menghemat sebuah kanal video menjadi sekitar 30 Kbps dan kanal suara menjadi 6 Kbps (*half-duplex*), artinya sebuah saluran *Local Area Network (LAN)* yang memiliki *bandwidth* sebesar 10/100 Mbps sebetulnya dapat digunakan untuk menyalurkan video dan audio sekaligus. Tentunya untuk kebutuhan konferensi yang *multiuser* akan dibutuhkan multi *bandwidth* pula, artinya minimal sekali kita harus menggunakan kanal 32-36 Kbps dikalikan dengan berapa *user* konferensi yang dilakukan dalam jaringan.

Pada Tabel 2.2 dibawah ini adalah standarisasi yang dikeluarkan oleh *ITU-T* untuk aplikasi kontrol audio, video dan *multiplex* pada jaringan yang mendukung sistem standar komunikasi video dan audio secara *realtime*.

Tabel 2.2 *ITU-T Multimedia Conferencing Standar* (*Basic Modes*) [15]

Standar	Network	Video	Audio	Multiplek	Control
H.320(1990)	ISDN	H.261	G.711	H.221	H.242
H.321(1995)	ATM/BISDN	Adapts H.320 to ATM/B-ISDN network			
H.322(1995)	IsoEthernet	Adapts H.320 to IsoEthernet Network			
H.323(1996)	LAN/internet	H.261	G.711	H.225.0	H.245
H.324(1995)	PSTN	H.263	G.723.1	H.223	H.245
H.310(1996)	ATM/BISDN	H.262	MPEG-1	H.222	H.245

BAB 3

PERANCANGAN DAN IMPLEMENTASI JARINGAN *VPN* DENGAN *ROUTING PROTOCOL RIPv2*, *EIGRP* DAN *OSPF*

Telah dijelaskan dalam Bab 2 tentang dasar teori yang mendukung pembuatan tugas akhir ini. Pada Bab 3 akan dijelaskan dengan lebih spesifik mengenai rancangan pemodelan sistem yang dibuat dan *tools* yang digunakan.

3.1 Spesifikasi dan Perancangan Sistem

Pada tugas akhir ini akan dibangun simple *VPN network* menggunakan teknologi *MPLS* dan *IPSec* mode *Tunnel*. Topologi jaringan dibangun dengan skenario *Site-to-site VPN Business*.

3.1.1 Kebutuhan *Hardware*

Dalam pembuatan tugas akhir ini sistem yang akan dibangun berupa testbed, jaringan dalam skala kecil, terdiri atas 5 buah router, 1 buah switch dan 4 buah PC dengan spesifikasi sebagai berikut :

- 3 buah Router cisco tipe 2600 *series*, 2 PE dan 1 *Core*
  - ✓ Tipe router : Cisco 2621 (MPC860)
  - ✓ *IOS version* : 12.3(24a)
- 2 buah Router cisco tipe 800 *series*, untuk 2 CE
  - ✓ Tipe router : Cisco 871 (MPC8272)
  - ✓ *IOS version* : 12.4(4)T7
- 1 buah Switch Cisco Catalyst tipe 2950, berfungsi sebagai hub, untuk menambah port Ethernet di router *PE*, agar *PC* sniffer dapat terhubung untuk memastikan paket data yang lewat dari *PE* ke *CE* dan atau sebaliknya.
- 4 buah *PC* dengan rincian sebagai sebagai berikut :
  - ➢ 1 *PC server*, spesifikasi :
    - *Processor* : Intel *Core* 2 Duo T8100 @2.10 GHz, 2094 MHz
    - *Memory* : DDR2 SDRAM 2 GB
    - Sistem Operasi : Linux CentOS 5.8

- *Integrated camera*

➢ 2 *PC client*, spesifikasi sebagai berikut :

PC 1 :

- *Processor* : Intel(R) Atom(TM) CPU N270 @1.60GHz 797 MHz
- *Memory* : 1.99 GB
- Sistem Operasi : Microsoft Windows XP Home Edition SP3
- *Integrated camera*

PC 2 :

- *Processor* : Intel(R) Atom(TM) CPU N280 @1.66GHz 1.66 GHz
- *Memory* : 0.99 GHz
- Sistem Operasi : Microsoft Windows XP Home Edition SP3
- *Integrated camera*

➢ 1 buah PC dengan wireshark untuk menangkap data yang lewat

- 2 buah kabel *UTP* tipe *crossover*, untuk menghubungkan router *PE – Core* dalam *cloud MPLS*
- 6 buah kabel *UTP* tipe *straight*, untuk menghubungkan router *PE – CE* dan *CE – PC* user
- Kabel console serial to USB
- Headset

3.1.2 Kebutuhan *Software*

Software yang digunakan untuk merealisasikan sistem ini antara lain :

1) Wireshark

Wireshark merupakan *software* yang digunakan untuk melakukan analisa parameter-parameter jaringan komputer. Wireshark dapat menangkap protokol yang sedang berjalan dalam suatu jaringan dan menganalisa parameter-parameternya yang menunjukkan nilai *QoS* dari jaringan tersebut, seperti *delay, jitter, throughput,* dan *packet loss.* Versi wireshark yang digunakan untuk

pengujian ini adalah wireshark 1.2.3 dan dapat di-*download* secara gratis pada website www.wireshark.org.

2) TrixBox

Trixbox (sebelumnya Asterisk@home) adalah sebuah VoIP *phone system* yang mudah diinstalasi, memanfaatkan *software* Asterisk PBX sebagai fungsi utama. Didesain baik untuk kalangan instansi perkantoran ataupun pengguna pribadi dengan menyertakan CentOS Linux, Sugar CRM, MySQL, HUDlite, dan *free* PBX yang menjadi satu *bundle* aplikasi. Trixbox muncul karena ada keluhan dari banyak *user* yang merasa kesulitan untuk mengaplikasi teknologi *VoIP* ini karena memerlukan usaha yang besar bahkan harus menjadi seorang programmer disebabkan oleh masalah user *interface* yang dirasa tidak *user friendly*. Oleh karena itu Trixbox diluncurkan guna mengatasi masalah tersebut. Trixbox dapat dikonfigurasi untuk menangani mulai dari satu sambungan telepon pribadi, puluhan bahkan ratusan sambungan telepon untuk perkantoran, dan dapat disambungkan ke beberapa saluran T1 di sebuah *call center* yang menangani jutaan menit percakapan per bulan. Sebuah web *GUI* (*web based*) memungkinkan konfigurasi dan pengoperasian menjadi mudah.

3) EyeBeam *softphone*

Gambar 3.1 Tampilan EyeBeam *softphone* saat melakukan *video conference*

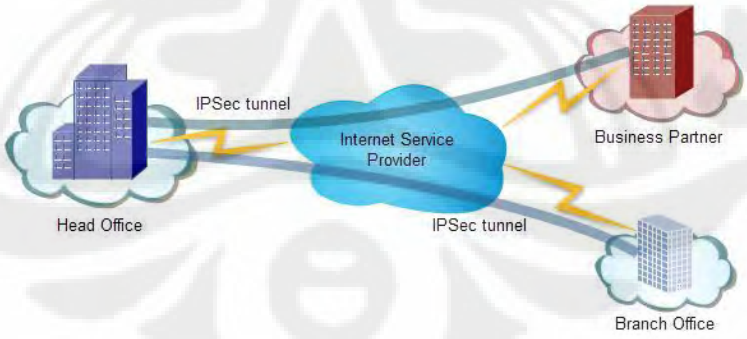
EyeBeam merupakan *softphone* yang berfungsi sebagai *User Agent Client (UAC)* yang bertugas untuk memulai sesi komunikasi dan *User Agent Server (UAS)* yang bertugas menagggapi atau menerima sesi komunikasi pada

protokol *Session Initition Protocol* (*SIP).* EyeBeam digunakan karena memiliki beberapa kelebihan, yaitu selain mendukung protocol *SIP* dan H.323, EyeBeam juga mendukung berbagai jenis kompresi *codec*, *video conference*, metode keamanan *Secure Real-time Protocol* (*SRTP)* dan *TLS*.

3.2 Pemodelan Sistem

Dianalogikan terdapat sebuah perusahaan A yang mempunyai banyak kantor cabang dan mitra bisnis, juga terdapat sebuah *provider* yang menyewakan layanan komunikasi data *via* internet menggunakan teknologi *MPLS*. Perusahaan A adalah salah satu *customer* dari p*rovider* tersebut. Setiap kantor cabang atau mitra bisnis dari perusahaan A diharapkan dapat mengakses sumberdaya yang ada di kantor pusat dengan aman dan cepat, bahkan dapat melakukan meeting atau diskusi jarak jauh antara user di kantor pusat dengan user di kantor cabang atau kantor pusat, atau bahkan dengan mitra bisnisnya dengan *via video conference*, sehingga dapat menghemat tenaga, waktu dan biaya.

Selain teknologi *MPLS* disisi *provider*, *IPSec* diimplementasikan pada masing-masing router *Customer Edge (CE)* untuk menjamin keamanan koneksi *end-to-end* router, dengan asumsi bahwa *Local Area Network* perusahaan A adalah jaringan yang secure/terpercaya. Perancangan sistem ini dibuat dengan skenario "*Site-to-site VPN Business*" seperti yang terlihat pada Gambar 3.2.

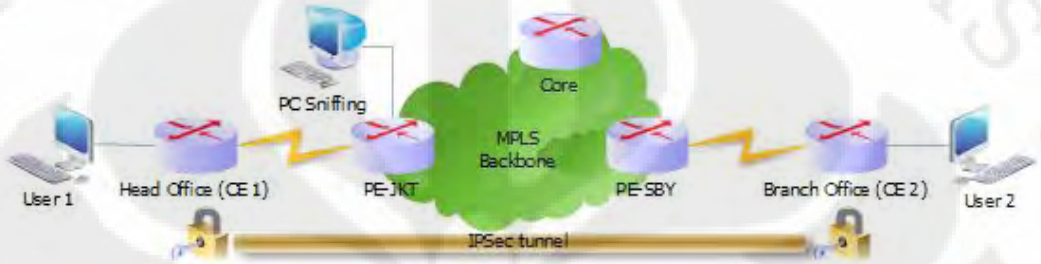


Gambar 3.2 Gambaran *Site-to-site VPN Business*

Pada dasarnya sebuah jaringan *MPLS* terdiri atas dua lapisan, yaitu :

1) Lapisan inti, terdiri atas interkoneksi sesama router *Core*
2) Lapisan luar, terdiri atas router *Provider Edge (PE)* yang mengitari lapisan inti

Pada tugas akhir ini, jaringan *VPN* yang dibangun terdiri atas 1 buah router *Core*, 2 buah router *PE* dan 2 buah router *CE*. Salah satu router *PE* akan diasumsikan berlokasi di kota Jakarta dan lainnya di Surabaya, kedua router *PE* tersebut menghubungkan router *CE* yang juga berlokasi di Jakarta untuk *Head Office* (kantor pusat) dan di Surabaya untuk *Branch Office* (kantor cabang) dari perusahaan A. Topologi jaringan *VPN* yang akan dibangun dapat dilihat pada Gambar 3.3.



Gambar 3.3 Topologi jaringan *VPN* berbasis *MPLS* dan *IPSec*

Untuk membangun jaringan *VPN* ini dibagi menjadi dua tahap, yaitu :

1) Membangun jaringan *MPLS* backbone disisi provider, dengan melakukan konfigurasi router PE-JKT , *Core* dan PE-SBY. Protokol *routing* yang digunakan yaitu *MP-BGP* untuk *peering* antar PE, dan *RIPv2*, *EIGRP* atau *OSPF* untuk koneksi antara *PE - Core*.
2) Membentuk koneksi *VPN* dengan tunnel *IPSec*, melakukan konfigurasi pada kedua router *CE* atau *end-to-end* router *Head Office* dan *Branch Office* dengan menggunakan *routing* protokol *RIPv2*, *EIGRP* atau *OSPF*.

3.3 Implementasi Sistem

Rencana implementasi merupakan tahap awal dari penerapan sistem yang baru dirancang. Implementasi sistem ini terdiri atas 3 skenario, dimana pada setiap skenario diimplementasikan *routing protocol* yang berbeda kemudian dilakukan pengujian dan pengambilan data untuk mengetahui seberapa besar pengaruh algoritma *routing* tersebut terhadap *QoS* dan kualitas *video conference* yang dilakukan karena implementasi *IPSec* pada *end-to-end* router di dalam jaringan *MPLS*, diharapkan agar sistem yang dibuat dapat beroperasi dan berjalan sesuai dengan yang diharapkan.

Daftar *interface* serta *IP Addres* yang akan digunakan adalah sebagai berikut :

Tabel 3.1 Konfigurasi IP pada masing-masing *interface* router *CE*

Device	Loopback0	Fastethernet0 (Fa0)	Fastethernet4 (Fa4)
Head Office	10.10.12.1/32	10.10.10.1/24	10.10.11.2/30
Branch Office	10.10.15.1/32	10.10.14.1/24	10.10.13.2/30

Tabel 3.2 Konfigurasi IP pada masing-masing *interface* router *service provider*

Device	Loopback0	Fastethernet0/0 (Fa0/0)	Fastethernet0/1 (Fa0/1)
PE-JKT	200.20.20.100/32	200.20.20.1/30	10.10.11.1/24
Core	200.20.20.101/32	200.20.20.2/30	200.20.20.5/30
PE-SBY	200.20.20.102/32	10.10.13.1/24	200.20.20.6/30

3.3.1 Skenario 1 – *RIPv2*

RIPv2 akan diimplemetasikan pada koneksi *Head Office* – PE-JKT, *Branch Office* - PE-SBY dan di dalam *cloud MPLS* (PE-JKT – *Core* – PE-SBY). Masing-masing skenario dibagi menjadi 2 tahap, dimana tahap pertama dilakukan ujicoba komunikasi *video conference* dengan dua user yang terhubung melalui jaringan *MPLS* dan tahap kedua dilakukan komunikasi *video conference* dengan 3 user melalui jaringan yang sama. *Bandwidth* yang diterapkan pada jaringan ini adalah *bandwidth* yang paling minimum untuk *video conference* dengan kombinasi *video codec* dan *audio codec* yang sengaja dipilih agar dengan *bandwidth* minimum dapat diperoleh kualitas *video conference* yang baik juga. *Video codec* dan *audio codec* yang digunakan adalah H.263 dan G.711, dengan *bandwidth* 256 Kbps.

3.3.1.1 Implementasi Jaringan *MPLS Backbone*

Langkah-langkah implementasi jaringan *MPLS backbone* ini, antara lain :

1) Memberikan *IP Address* pada *interface* router

   IP Address dikonfigurasi pada *interface* kearah *Core* di router PE-JKT dan PE-SBY, sedangkan *IP Address* pada *interface* yang kearah CE (*Head Office* dan *Branch Office*) akan dikonfigurasi setelah membuat konfigurasi router *virtual* di *interface* tersebut.

Konfigurasi di router PE-JKT :

int f0/0
IP Address 200.20.20.1 255.255.255.252

Konfigurasi di router *Core* :

int f0/0
IP Address 200.20.20.2 255.255.255.252
int f0/1
IP Address 200.20.20.9 255.255.255.252

Konfigurasi di router PE-SBY :

int f0/1
IP Address 200.20.20.10 255.255.255.252

Langkah ini dimaksudkan untuk memastikan kondisi link antara *PE* dan *Core*, yaitu dengan cara melakukan tes ping ke masing-masing *interface* yang berada dalam satu *network*, jika berhasil maka artinya link antara *PE* dan *Core* sudah bagus.

2) Membuat router *virtual*

Provider yang mempunyai banyak *customer* dari berbagai perusahaan memerlukan router - router *virtual* di setiap router *PE* yang terhubung langsung dengan router *customer (CE),* dimana router *virtual* tersebut terdiri atas *Route Distinguisher (RD)* dan *Virtual Routing Forwarding (VRF)*. *Route distinguisher* merupakan proses pemberian label dari setiap *VPN customer* yang nantinya akan dipertukarkan antar router PE, sedangkan nama dari Router *virtual* ini diasumsikan sebagai mini router yang *dedicated* untuk mengatur tabel *routing* setiap *customer*. Karena konsepnya *dedicated*, maka jumlah router *virtual* ini akan sebanyak jumlah *VPN customer* yang terhubung dengan *PE* tersebut.

Dalam tugas akhir ini terdapat satu buah *VPN customer* yaitu *Company* A, sehingga hanya terdapat satu buah router *virtual* didalam fisik router PE-JKT dan PE-SBY.

Konfigurasi router *virtual* untuk PE-JKT dan PE-SBY :

ip vrf company-a
rd 65000:1
route-target export 65000:1
route-target import 65000:1

Setelah terbentuk router *virtual* yang ber-ID company-a, selanjutnya mengimplementasikan router *virtual* tersebut ke *interface* di router PE yang kearah router *CE* dan memberikan *IP Address* pada *interface* tersebut setelahnya.

Konfigurasi di router PE-JKT :

int fa0/1
ip vrf forwarding company-a
IP Address 10.10.11.1 255.255.255.252

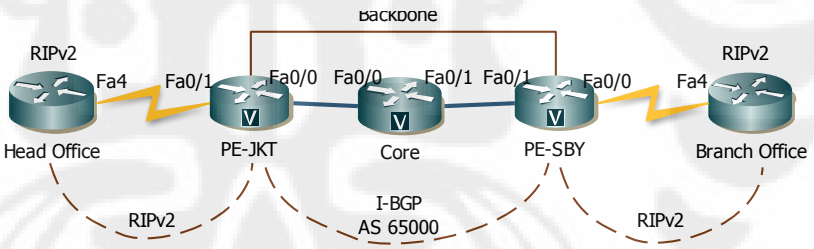
Contoh konfigurasi di router PE-SBY :

int fa0/0
ip vrf forwarding company-a
IP Address 10.10.13.1 255.255.255.252

3) Mengaktifkan *routing* dinamik

Untuk membuat *cloud MPLS*, semua router dalam *cloud* tersebut harus menggunakan *routing* dinamik, agar *routing* di masing-masing router dapat mengembang atau terpopulasi.

a) Mengaktifkan *RIPv2*



Gambar 3.4 Jaringan *VPN* menggunakan *routing protocol RIPv2*

Konfigurasi *RIPv2* pada router PE-JKT, PE-SBY dan *Core* :

router rip
version 2
network 200.20.20.0

Network yang akan di-*advertise* adalah *interface* Loopback0 dan *network* dari Ethernet yang *directly connected.* Dalam konteks PE-JKT, *interface* yang perlu di-*advertise* adalah *network* dari Fastethernet0/0 yang mengarah ke *Core*. Sedangkan *interface* Fastethernet0/1 yang mengarah ke *CE Head Office* tidak perlu di-*advertise*. Hal ini dimaksudkan agar didalam *cloud MPLS*, *IP*

prefix yang di-*advertise* hanyalah yang benar-benar berada didalam *cloud MPLS*. Router *Core* meng-*advertise interface* yang kearah router PE-JKT dan PE-SBY. Semua *interface* yang akan diadvertise dalam *cloud MPLS* ini menggunakan kelas IP yang sama yaitu kelas C, sedangkan karakteristik *RIPv2* salah satunya adalah mendukung *Variable Length Subnet Mask* (*VLSM*) yaitu teknik untuk menghemat penggunaan *IP Address*, yang menjadikan *routing* ini sebagai salah satu *classless routing protocol*, sehingga dalam konfigurasi router *RIP network* yang akan di-*advertise* cukup di-*set* subnet-nya saja "200.20.20.0".

b) Mengaktifkan *BGP*

Untuk mengaktifkan *BGP* maka status *BGP connection* antara PE-JKT dan PE-SBY harus *established* terlebih dahulu.

Konfigurasi di PE-JKT :

```
router BGP 65000
 no synchronization
 BGP log-neighbor-changes
 neighbor 200.20.20.102 remote-as 65000
 neighbor 200.20.20.102 update-source Loopback0
 neighbor 200.20.20.102 next-hop-self
 no auto-summary
```

Konfigurasi di PE-SBY :

```
router BGP 65000
 no synchronization
 BGP log-neighbor-changes
 neighbor 200.20.20.100 remote-as 65000
 neighbor 200.20.20.100 update-source Loopback0
 neighbor 200.20.20.100 next-hop-self
 no auto-summary
```

Untuk memastikan *TCP connetion* antara *BGP* yang ada di router PE-JKT dan PE-SBY sudah *established* atau belum, dapat diperiksa di salah satu router *PE* seperti dibawah ini :

```
PE-JKT#sh ip BGP sum
BGP router identifier 200.20.20.100, local AS number 65000
BGP table version is 1, main routing table version 1
Neighbor        V    AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
200.20.20.102   4 65000     132     132        1    0    0 02:05:02        0
```

Dari hasil diatas terlihat bahwa status *BGP* sudah establish, tertulis "02:05:02" artinya *BGP* sudah establish selama 2 jam 5 menit 2 detik.

c) Mengaktifkan *MP-BGP*

Koneksi *MP-BGP* antara PE-JKT dan PE-SBY disebut sebagai *Backbone MP-BGP*. Setelah mengaktifkan *BGP* antara PE-JKT dan PE-SBY, *peering* (bisa disebut sebagai *tunnel*) *BGP* yang telah terbentuk akan berisi sub tunnel *MP-BGP* untuk membawa tabel *routing* vrf company-a dari PE-JKT ke PE-SBY dan sebaliknya.

Konfigurasi di PE-JKT :

```
router BGP 65000
...
address-family VPNv4
 neighbor 200.20.20.102 activate
 neighbor 200.20.20.102 send-community both
 exit-address-family
```

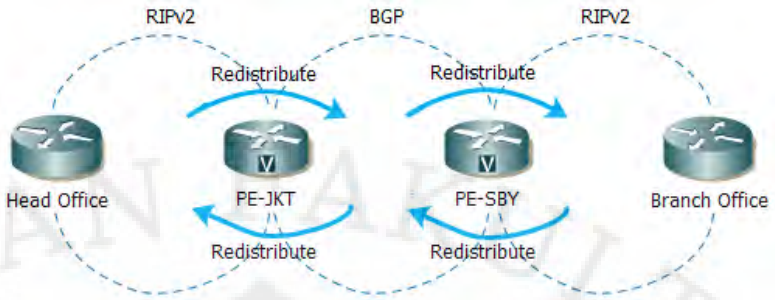
Konfigurasi di PE-SBY :

```
router bgo 65000
…
address-family VPNv4
 neighbor 200.20.20.100 activate
 neighbor 200.20.20.100 send-community both
 exit-address-family
```

Konfigurasi diatas seolah-olah membuka sebuah *sub tunnel* di *tunnel BGP* AS number 65000 yang sebelunnya sudah ada, yaitu sub tunnel untuk melewatkan tabel *routing* vrf company-a. *MP-BGP* memanfaatkan fasilitas *community* pada standar *BGP*, baik yang *standard* maupun *extended*, dimana keduanya *(both)* akan saling dipertukarkan.

d) Mengaktifkan *RIPv2* pada router *virtual* dan me-*redistribute routing RIPv2* ke dalam *MP-BGP* dan *MP-BGP* ke dalam *RIPv2*

Routing protocol yang digunakan pada jaringan ini dapat digambarkan sebagai *RIPv2-BGP-RIPv2* seperti yang terlihat pada Gambar 3.5. Dengan demikian diperlukan suatu proses *redistribution*, yaitu proses pada router untuk meng-*import* atau meng-*export routing* protokol satu ke *routing* protokol yang lain agar kedua *routing* protokol yang berbeda tersebut dapat saling berkomunikasi.



Gambar 3.5 Ilustrasi *redistribute network RIPv2* dan *BGP*

Konfigurasi di PE-JKT dan PE-SBY :

Redistribute dari RIP ke *MP-BGP*

```
address-family ipv4 vrf company-a
 redistribute BGP 65000 metrik transparent
 network 10.0.0.0
 no auto-summary
 version 2
 exit-address-family
```

Redistribute dari *MP-BGP* ke *RIPv2*

```
address-family ipv4 vrf company-a
 redistribute rip
 no auto-summary
 no synchronization
 exit-address-family
```

4) Mengaktifkan *MPLS*

Agar *MPLS* aktif di PE-JKT, *Core* dan PE-SBY, maka *interface* di router tersebut yang menerapkan label harus diaktifkan *MPLS*-nya terlebih dahulu.

Mengaktifkan *MPLS* di PE-JKT :

```
interface FastEthernet0/0
description "Connection to CORE"
 ...
 tag-switching ip
```

Mengaktifkan *MPLS* di *Core* :

```
interface FastEthernet0/0
description "Connection to PE-JKT"
...
tag-switching ip
```

Mengaktifkan *MPLS* di PE-SBY :

```
interface FastEthernet0/1
```

```
description "Connection to CORE"
...
tag-switching ip
```

Untuk memastikan bahwa *MPLS* sudah bekerja, dapat di periksa di salah satu router tersebut seperti dibawah ini :

```
CORE#sh MPLS ldp neighbor
   Peer TDP Ident: 200.20.20.100:0; Local TDP Ident 200.20.20.101:0
     TCP connection: 200.20.20.100.711 - 200.20.20.101.50374
     State: Oper; PIEs sent/rcvd: 0/14; Downstream
     Up time: 00:08:30
     TDP discovery sources:
       FastEthernet0/0, Src IP addr: 200.20.20.1
     Addresses bound to peer TDP Ident:
       200.20.20.1     200.20.20.100
   Peer TDP Ident: 200.20.20.102:0; Local TDP Ident 200.20.20.101:0
     TCP connection: 200.20.20.102.55541 - 200.20.20.101.711
     State: Oper; PIEs sent/rcvd: 0/13; Downstream
     Up time: 00:08:23
     TDP discovery sources:
       FastEthernet0/1, Src IP addr: 200.20.20.10
     Addresses bound to peer TDP Ident:
       200.20.20.10    200.20.20.102
```

Dari hasil pengecekan diatas terlihat bahwa router *Core* sudah mengenali router PE-JKT dan PE-SBY sebagai router *MPLS*, artinya *MPLS* sudah aktif di ketiga router tersebut.

3.3.1.2 Implementasi Router *CE*

Langkah-langkah konfigurasi router *CE* :

1) Memberikan *IP Address* pada semua *interface* router

   Konfigurasi router *Head Office* :

```
interface Loopback0
 IP Address 10.10.12.1 255.255.255.255

interface FastEthernet0
description “Connection to PC-User1”
 switchport access vlan 2

interface FastEthernet1
 switchport access vlan 2

interface FastEthernet4
 description "Connetion to PE-JKT"
 IP Address 10.10.11.2 255.255.255.252

interface Vlan2
 IP Address 10.10.10.1 255.255.255.0
```

PC-User1 yang terhubung ke Fastethernet0 bertindak sebagai *server* Tribox dan juga *client.*

Konfigurasi router *Branch Office* :

```
interface Loopback0
 IP Address 10.10.15.1 255.255.255.255

interface FastEthernet0
description "Connection to PC-User2"
 switchport access vlan 2

interface FastEthernet1
description "Connection to PC-User3"
 switchport access vlan 2

interface FastEthernet2
description "Connection to PC-Wireshark"
 switchport access vlan 2

interface FastEthernet4
 description "Connetion to PE-JKT"
 IP Address 10.10.13.2 255.255.255.252

interface Vlan2
 IP Address 10.10.14.1 255.255.255.0
```

PC-User2 yang terhubung ke Fastethernet0 dan PC-User3 yang terhubung ke fastethernet1 bertindak sebagai *UAC*, sedangkan PC-Wireshark yang terhubung ke Fastethernet2 bertindak sebagai *PC sniffing* untuk menangkap paket *RTP* yang melintasi jaringan.

2) Mengaktifkan *routing RIPv2*

Untuk mengaktifkan *routing RIPv2*, konfigurasi di router *Head Office* dan *Branch Office* adalah :

```
router rip
version 2
network 10.0.0.0
```

Network yang akan diadvertise pada router *Head Office* dan *Branch Office* berasal dari kelas *IP* yang sama, yaitu kelas A, sehingga *network* yang di-set pada router *RIP* adalah subnet dari IP tersebut "10.0.0.0".

3.3.1.3 Pengecekan *Routing Protocol RIPv2*

Untuk memastikan konfigurasi yang diimplementasikan sudah sesuai dengan yang diharapkan, maka dilakukan pengecekan *routing* pada masing-masing router, baik di router *CE* maupun di dalam *cloud MPLS*.

Head Office#sh ip route
CoDES: C - connected, S - static, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2
i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
*ia - IS-IS inter area, * - candidate default, U - per-user static route*
o - ODR, P - periodic downloaded static route
Gateway of last resort is not set
10.0.0.0/8 is variably subnetted, 6 subnets, 3 masks
C 10.10.10.0/24 is directly connected, Vlan2
C 10.10.11.0/30 is directly connected, FastEthernet4
R 10.10.13.0/30 [120/1] via 10.10.11.1, 00:00:02, FastEthernet4
C 10.10.12.1/32 is directly connected, Loopback0
R 10.10.15.1/32 [120/2] via 10.10.11.1, 00:00:02, FastEthernet4
R 10.10.14.0/24 [120/2] via 10.10.11.1, 00:00:02, FastEthernet4

PE-JKT#sh ip route
...
200.20.20.0/24 is variably subnetted, 5 subnets, 3 masks
C 200.20.20.100/32 is directly connected, Loopback0
R 200.20.20.101/32 [120/1] via 200.20.20.2, 00:00:27, FastEthernet0/0
R 200.20.20.102/32 [120/2] via 200.20.20.2, 00:00:27, FastEthernet0/0
R 200.20.20.8/30 [120/1] via 200.20.20.2, 00:00:27, FastEthernet0/0
C 200.20.20.0/29 is directly connected, FastEthernet0/0

CORE#sh ip rou
...
Gateway of last resort is not set
200.20.20.0/24 is variably subnetted, 5 subnets, 3 masks
R 200.20.20.100/32 [120/1] via 200.20.20.1, 00:00:03, FastEthernet0/0
C 200.20.20.101/32 is directly connected, Loopback0
R 200.20.20.102/32 [120/1] via 200.20.20.10, 00:00:07, FastEthernet0/1
C 200.20.20.8/30 is directly connected, FastEthernet0/1
C 200.20.20.0/29 is directly connected, FastEthernet0/0

Dari hasil pengecekan diatas terlihat bahwa *routing* protocol *RIPv2* sudah aktif pada semua *interface* router. Untuk memastikan *routing RIPv2* sudah aktif pada router *virtual* yang diimplementasikan pada PE-JKT dan PE-SBY dapat dicek pada salah satu *PE* sebagai berikut :

PE-JKT#sh ip rou vrf company-a
Routing Table: company-a
...
Gateway of last resort is not set
10.0.0.0/8 is variably subnetted, 6 subnets, 3 masks
R 10.10.10.0/24 [120/1] via 10.10.11.2, 00:00:09, FastEthernet0/1
C 10.10.11.0/30 is directly connected, FastEthernet0/1
B 10.10.13.0/30 [200/0] via 200.20.20.102, 02:04:07
R 10.10.12.1/32 [120/1] via 10.10.11.2, 00:00:09, FastEthernet0/1
B 10.10.15.1/32 [200/1] via 200.20.20.102, 02:04:38
B 10.10.14.0/24 [200/1] via 200.20.20.102, 02:04:38

Dari hasil pengecekan diatas dapat dilihat bahwa terdapat dua *routing* protokol yang aktif untuk vrf company-a pada PE-JKT yaitu *RIPv2* dan *MP-BGP*, dimana

MP-BGP bertugas membawa tabel *routing RIPv2* company-a melintasi *MPLS backbone.*

3.3.1.4 Implementasi *IPSec* Mode *Tunnel* di router *CE*

Untuk mengaktifkan enkripsi dan tunnel *IPSec* harus memenuhi langkah-langkah sebagai berikut :

1) Mengkonfigurasi *IKE policies*

   Internet Key Exchange (IKE) aktif secara default dan tidak harus diaktifkan pada masing-masing *interface* tetapi aktif secara global pada semua *interface* router. *IKE policies* harus dipasang pada setiap peer, dalam hal ini router *CE*. *IKE policies* didefinisikan sebagai sebuah kombinasi dari parameter-parameter *security* yang digunakan selama negosiasi *IKE*. *IKE policies* dapat dibuat lebih dari 1, dimana setiap *policies* memiliki kombinasi nilai parameter yang berbeda. Masing-masing *policies* yang dibuat diberi nilai prioritas yang unik yaitu dari 1 sampai 10000 (1 adalah prioritas tertinggi). Setiap *peer* dapat memiliki banyak *policies*, tetapi sedikitnya 1 *policies* harus mempunyai nilai parameter enkripsi, hash, autentikasi dan Diffie-Helman yang sama dengan salah satu *policies* yang ada pada remote peer.

   Konfigurasi di router *CE Head Office* dan *Branch Office* :

   ```
   crypto isakmp policy 1
    encr 3des
    authentication pre-share
    group 2
   crypto isakmp key p4ssw0rd address 10.10.11.2
   ```

   Sesuai dengan konfigurasi diatas, pada tugas akhir ini algoritma enkripsi yang digunakan adalah 168-bit *Tripple DES* (3*DES*), algoritma hash adalah sha, autentikasi menggunakan pre-shared dan Diffie-Hellman Identifier adalah 1024 bit (*group* 2).

2) Memeriksa *IKE policies*

   Pengecekan *IKE policies* dilakukan di kedua router *CE* dimana *IPSec* diimplementasikan. Kedua router *CE* harus mempunyai parameter enkripsi yang sama, yaitu akgoritma enkripsi, algoritma hash, metode autentikasi dan *Diffie-Hellam identifier/group*.

Pengecekan *IKE policies* di router *Head Office* :

Head Office#sh crypto isakmp policy
Global IKE policy
Protection suite of priority 1
encryption algorithm: Three key triple DES
hash algorithm: Secure Hash Standard
authentication method: Pre-Shared Key
Diffie-Hellman group: #2 (1024 bit)
lifetime: 86400 seconds, no volume limit
Default protection suite
encryption algorithm: DES - Data Encryption Standard (56 bit keys).
hash algorithm: Secure Hash Standard
authentication method: Rivest-Shamir-Adleman Signature
Diffie-Hellman group: #1 (768 bit)
lifetime: 86400 seconds, no volume limit

Pengecekan *IKE policies* di router *Branch Office* :

Branch Office#sh crypto isakmp policy
Global IKE policy
Protection suite of priority 1
encryption algorithm: Three key triple DES
hash algorithm: Secure Hash Standard
authentication method: Pre-Shared Key
Diffie-Hellman group: #2 (1024 bit)
lifetime: 86400 seconds, no volume limit
Default protection suite
encryption algorithm: DES - Data Encryption Standard (56 bit keys).
hash algorithm: Secure Hash Standard
authentication method: Rivest-Shamir-Adleman Signature
Diffie-Hellman group: #1 (768 bit)
lifetime: 86400 seconds, no volume limit

IPSec menggunakan metode *Internet Key Exchange* (*IKE*) untuk melakukan negosiasi kunci dan menentukan algoritma protokol (*AH* dan atau *ESP)* yang digunakan. Seperti yang terlihat dari hasil pengecekan *IKE policies* diatas, penentuan algoritma *IKE* meliputi :

- Algoritma enkripsi : *Three key triple DES* (3*DES*)
- Algortima hash : *Secure Hash Standard* (*SHA*)
- Metode autentikasi : *Pre-Shared-Key*
- Diffie-Helman *group* : 2 (1024 bit)

3) Mengkonfigurasi *IPSec* dan *IPSec* mode *tunnel*

Setelah melakukan konfigurasi *IKE policies* pada masing-masing *peer*, selanjutnya adalah melakukan konfigurasi *IPSec* di masing-masing *peer* dengan langkah-langkah seperti berikut :

- Membuat *crypto acces lists*

Crypto accest lists digunakan untuk mendifinisikan trafik *IP* mana yang akan dilindungi atau tidak dilindungi oleh *crypto*. *Acces lists* ini tidak sama dengan *acces lists* biasa yang menentukan apakah trafik akan diteruskan atau diblok di sebuah *interface*.

Konfigurasi di router *Head Office* :

ip access-list extended JKT-to-SBY
permit ip 10.10.10.0 0.0.0.255 10.10.14.0 0.0.0.255

Konfigurasi di router *Branch Office* :

ip access-list extended SBY-to-JKT
permit ip 10.10.14.0 0.0.0.255 10.10.10.0 0.0.0.255

Extended access list yang diberi nama JKT-to-SBY di router *Head Office* dan SBY-to-JKT di router *Branch Office* nantinya akan dipasang pada *interface* yang akan dilewati trafik *IPSec*.

- Memeriksa *crypto acces lists*

Branch Office#sh access-lists
Extended IP access list SBY-to-JKT
10 permit ip 10.10.14.0 0.0.0.255 10.10.10.0 0.0.0.255
***Head Office*#sh access-lists**
Extended IP access list JKT-to-SBY
10 permit ip 10.10.10.0 0.0.0.255 10.10.14.0 0.0.0.255

Trafik data akan diproteksi dari router *Branch Office* hingga *Head Office,* demikian sebaliknya.

- Mendefinisikan *transform-set* dan mengkonfigurasi *IPSec tunnel mode*

Transform-set harus didefinisikan sesuai dengan protokol yang digunakan.

Konfigurasi di router *Head Office* dan *Branch Office* :

crypto IPSec transform-set finalproject esp-3DES esp-sha-hmac

Dilihat dari konfigurasi diatas, sesuai dengan *IKE policies* yang telah diimplementasikan, *transform-set* diberi nama finalproject dan terdiri atas kombinasi esp-sha-hmac, sedangkan *encryption set* terdiri atas kombinasi esp-3des, *ESP* diimplementasikan pada *SA* (*Security Association*) mode *tunnel,* dimana proteksi hanya diberikan pada paket yang berada di dalam tunnel. *ESP* dibuat

dengan melakukan enkripsi pada paket IP dan membuat paket IP lain yang mengandung *header* IP asli dan *header ESP*.

- Memeriksa *transform-set* dan *IPSec tunnel mode*

  Pengecekan disalah satu router *Branch Office* maupun *Head Office* diperoleh hasil yang sama sebagai berikut :

  Branch Office*#sh *crypto IPSec transform-set
  ***Transform-set* finalproject: { esp-3*DES* esp-sha-hmac }**
  will negotiate = { Tunnel, },

  IPSec diimplementasikan dalam mode Tunnel, dengan *transform-set* yang telah didefinisikan dengan nama "finalproject".

4) Mengkonfigurasi *crypto maps*

   Ketika kedua *peer* mencoba membangun sebuah *Security Association (SA),* maka masing-masing dari peer tersebut harus mempunyai sedikitnya satu daftar *crypto maps* yang sesuai dengan salah satu daftar *crypto maps* pada *peer* yang lainnya. Langkah-langkah untuk melakukan konfigurasi *crypto maps* antara lain :

   - Membuat daftar *crypto maps*

     Daftar *crypto maps* ini menggunakan *IKE* untuk membangun *SA*.

     Konfigurasi *crypto maps* di router *Head Office* :

     crypto maps skripsi 1 IPSec-isakmp
     set peer 10.10.13.2
     set transform-set finalproject
     match address JKT-to-SBY

     Konfigurasi *crypto maps* di router *Branch Office* :

     crypto maps skripsi 1 IPSec-isakmp
     set peer 10.10.11.2
     set transform-set finalproject
     match address SBY-to-JKT

     Crypto maps yang diimplementasikan diberi nama "skripsi", set peer adalah *interface* peer lawan yang juga mengimplementasikan *IPSec*, *transform-set* "finalproject" dan *extended access lists* yang telah dibuat diimplementasikan dalam daftar *crypto maps*.

   - Memeriksa daftar *crypto maps*

     Pengecekan daftar *crypto maps* disalah satu router *CE :*

```
Branch-Office#sh crypto map
Crypto Map "skripsi" 1 IPSec-isakmp
      Peer = 10.10.11.2
      Extended IP access list SBY-to-JKT
          access-list SBY-to-JKT permit ip 10.10.14.0 0.0.0.255 10.10.10.0
0.0.0.255
      Current peer: 10.10.11.2
      Security association lifetime: 4608000 kilobytes/3600 seconds
      PFS (Y/N): N
      Transform sets={
            finalproject,
      }
      Interfaces using crypto map skripsi:
            FastEthernet4
```

Crypto maps dengan nama "skripsi" mempunyai *peer* dengan IP Address 10.10.11.2 yaitu router *Head Office* (interface kearah PE-JKT atau diasumsikan sebagai *interface WAN*).

- Mengimplementasikan *crypto maps* ke *interface*

  Crypto maps diimplementasikan pada masing-masing *interface* yang dilewati trafik *IPSec*. Implementasi *crypto maps* pada sebuah *interface* router membuat router mengevaluasi semua trafik pada *interface* sesuai dengan daftar *crypto maps* dan menggunakan *policies* tertentu selama negosiasi *SA* agar trafik mendapatkan proteksi dari *crypto*.

  Konfigurasi *crypto maps* di router *Head Office* :

```
interface FastEthernet4
description "Connetion to PE-JKT"
...
crypto maps skripsi
```

  Konfigurasi *crypto maps* di router *Head Office* :

```
interface FastEthernet4
description "Connetion to PE-SBY"
...
crypto maps skripsi
```

- Memeriksa *crypto maps interface associations*

  Pemeriksaan implementasi *crypto maps* pada salah satu *interface* router *CE* kearah *PE* dapat dilakukan sebagai berikut :

```
Branch-Office#sh crypto map interface fastEthernet 4
Crypto Map "skripsi" 1 IPSec-isakmp
      Peer = 10.10.11.2
      Extended IP access list SBY-to-JKT
          access-list SBY-to-JKT permit ip 10.10.14.0 0.0.0.255 10.10.10.0
0.0.0.255
      Current peer: 10.10.11.2
```

Security association lifetime: 4608000 kilobytes/3600 seconds
PFS (Y/N): N
Transform sets={
finalproject,
}
Interfaces using crypto map skripsi:
FastEthernet4

Crypto maps diimplementasikan pada *interface* Fastethernet4, yaitu *interface WAN* di router *CE*.

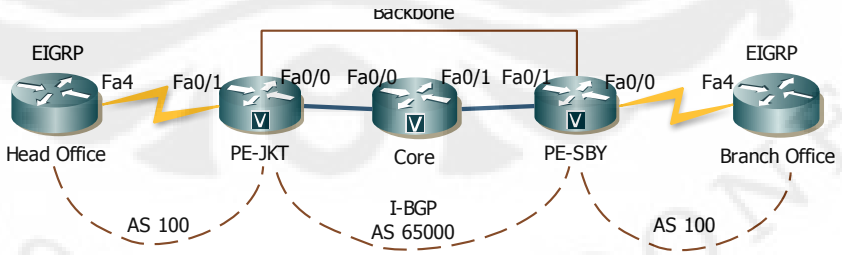
3.3.2 Skenario 2 – *EIGRP*

Sama halnya dengan skenario 1, terdapat 2 tahap pada skenario 2, yaitu ujicoba *video conference* dengan 2 *user* dan 3 *user*. Masing-masing akan dilakukan analisa pengaruh algoritma *routing* yang digunakan terhadap *IPSec* dalam jaringan *MPLS*. Jaringan yang dibangun sama, langkah-langkah implementasi *MPLS backbone* terdapat perbedaan pada langkah ke-3 yaitu mengaktifkan *routing* dinamik, dimana *RIPv2* diganti dengan *EIGRP*, langkah-langkah mengaktifkan router *CE* juga terdapat sedikit perbedaan yaitu pada langkah ke-2 "Mengaktifkan *routing RIPv2*", dimana pada skenario ini *RIPv2* diganti dengan *EIGRP*, sedangkan untuk implementasi *IPSec* langkah-langkah implementasi sama dengan pada skenario pertama.

3.3.2.1 Impelementasi *MPLS Backbone*

Langkah 1 s/d 2 sama dengan skenario pertama. Langkah-langkah untuk mengaktifkan *EIGRP* pada *cloud MPLS* antara lain :

- Mengaktifkan *EIGRP* di router PE-JKT, *Core* dan PE-SBY.



Gambar 3.6 Jaringan *VPN* menggunakan *routing* protokol *EIGRP*

Konfigurasi *EIGRP* pada router PE-JKT, PE-SBY dan *Core* :

router eigrp 100
network 200.20.20.0

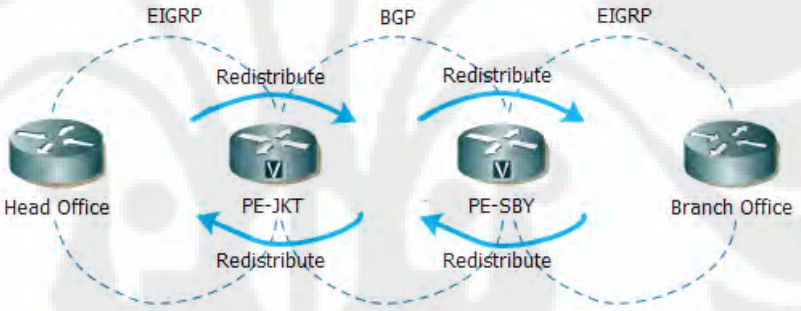
auto-summary

Dalam *EIGRP* dikenal istilah *Autonomous System* (AS) yang berfungsi untuk membuat semua router yang mempunyai *AS* sama berada dalam satu lingkup area sehingga dapat melakukan komunikasi data antara sumber dan tujuan. Seperti yang terlihat pada konfigurasi diatas bahwa semua router dalam jaringan ini dikonfigurasi dalam satu *AS* yaitu 100.

Sama halnya dengan *RIPv2*, *EIGRP* juga mendukung *VLSM* dan merupakan *classless routing*, sehingga *network* yang di-*set* pada konfigurasi router *EIGRP* cukup subnetnya saja.

Langkah-langkah untuk mengaktifkan *BGP* dan *MP-BGP* sama dengan skenario pertama.

- Mengaktifkan *EIGRP* pada router *virtual* dan me-*redistribute EIGRP* ke *BGP* dan *BGP* ke *EIGRP* di router PE-JKT dan PE-SBY



Gambar 3.7 Analogi *redistribute EIGRP* ke *BGP* dan *BGP* ke *EIGRP*

Redistribute EIGRP ke *BGP* :

```
router EIGRP 100
...
address-family ipv4 vrf company-a
 redistribute BGP 65000 metrik 10000 100 255 1 1500
 network 10.0.0.0
 auto-summary
 autonomous-system 100
 EIGRP router-id 200.20.20.100
 exit-address-family
```

Redistribute BGP ke *EIGRP* :

```
router BGP 65000
 ...
address-family ipv4 vrf company-a
 redistribute EIGRP 100
 no auto-summary
 no synchronization
```

exit-address-family

Untuk melakukan *redistribute* dari *EIGRP* ke *BGP* diperlukan definisi *metrik* dari link yang ada, dimana *EIGRP* memerlukan 5 *metric* ketika mendistribusikan protokol lain, seperti *bandwidth*, *delay*, kehandalan, beban dan *MTU*. Default *metric* yang di-*set* seperti konfigurasi diatas pada router PE-JKT dan PE-SBY berasal dari nilai berikut ;

- *Bandwidth*, dalam kilobit per detik (Kbps) adalah 10.000 untuk Ethernet.
- *Delay*, dalam puluhan mikrodetik (ms) adalah 100 x 10 = 1 ms.
- Kehandalan, 255 untuk 100% kehandalan.
- Beban, efektif beban pada link dinyatakan sebagai angka dari 0 hingga 255, dimana 255 adalah 100% loading.
- *Maximum Transmission Unit* (*MTU*), *minimum MTU* dari link yang ada biasanya sama dengan *interface* Ethernet, yaitu 1500 bytes.

3.3.2.2 Implementasi Router *CE*

Konfigurasi *IP Address* yang dipasang pada masing-masing *interface* router *Head Office* dan *Branch Office* sama dengan pada skenario pertama, sedangkan untuk mengaktifkan *EIGRP* di kedua router *CE* tersebut adalah sebagai berikut :

```
router EIGRP 100
network 10.0.0.0
auto-summary
```

Baik router *CE* semua berada dalam *AS* yang sama dengan router *PE* dan *Core*, yaitu 100, agar semua router tersebut dapat saling berkomunikasi.

3.3.2.3 Pengecekan *Routing Protocol EIGRP*

Untuk memeriksa *routing protocol* yang telah diimplementasikan, dapat dilakukan dengan perintah dibawah di masing-masing router baik di *cloud MPLS* maupun di router *CE* :

```
Head Office#sh ip rou
CoDES: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
```

E1 - OSPF external type 1, E2 - OSPF external type 2
i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
*ia - IS-IS inter area, * - candidate default, U - per-user static route*
o - ODR, P - periodic downloaded static route
Gateway of last resort is not set
10.0.0.0/8 is variably subnetted, 6 subnets, 3 masks
C 10.10.10.0/24 is directly connected, Vlan2
C 10.10.11.0/30 is directly connected, FastEthernet4
D 10.10.13.0/30 [90/30720] via 10.10.11.1, 00:21:12, FastEthernet4
C 10.10.12.1/32 is directly connected, Loopback0
D 10.10.15.1/32 [90/158720] via 10.10.11.1, 00:21:12, FastEthernet4
D 10.10.14.0/24 [90/33280] via 10.10.11.1, 00:21:12, FastEthernet4

PE-JKT# sh ip rou
...
Gateway of last resort is not set
200.20.20.0/24 is variably subnetted, 5 subnets, 3 masks
C 200.20.20.100/32 is directly connected, Loopback0
D 200.20.20.101/32
[90/156160] via 200.20.20.2, 00:19:50, FastEthernet0/0
D 200.20.20.102/32
[90/158720] via 200.20.20.2, 00:19:50, FastEthernet0/0
D 200.20.20.8/30 [90/30720] via 200.20.20.2, 00:19:50, FastEthernet0/0
C 200.20.20.0/29 is directly connected, FastEthernet0/0

CORE#sh ip rou
...
Gateway of last resort is not set
200.20.20.0/24 is variably subnetted, 5 subnets, 3 masks
D 200.20.20.100/32
[90/156160] via 200.20.20.1, 00:19:11, FastEthernet0/0
C 200.20.20.101/32 is directly connected, Loopback0
D 200.20.20.102/32
[90/156160] via 200.20.20.10, 00:19:11, FastEthernet0/1
C 200.20.20.8/30 is directly connected, FastEthernet0/1
C 200.20.20.0/29 is directly connected, FastEthernet0/0

PE-SBY#sh ip rou
...
Gateway of last resort is not set
200.20.20.0/24 is variably subnetted, 5 subnets, 3 masks
D 200.20.20.100/32
[90/158720] via 200.20.20.9, 00:17:54, FastEthernet0/1
D 200.20.20.101/32
[90/156160] via 200.20.20.9, 00:17:56, FastEthernet0/1
C 200.20.20.102/32 is directly connected, Loopback0
C 200.20.20.8/30 is directly connected, FastEthernet0/1
D 200.20.20.0/29 [90/30720] via 200.20.20.9, 00:17:56, FastEthernet0/1

Branch Office#sh ip rou
...
Gateway of last resort is not set
10.0.0.0/8 is variably subnetted, 6 subnets, 3 masks
D 10.10.10.0/24 [90/33280] via 10.10.13.1, 00:15:35, FastEthernet4
D 10.10.11.0/30 [90/30720] via 10.10.13.1, 00:15:35, FastEthernet4
C 10.10.13.0/30 is directly connected, FastEthernet4
D 10.10.12.1/32 [90/158720] via 10.10.13.1, 00:15:35, FastEthernet4

C 10.10.15.1/32 is directly connected, Loopback0
C 10.10.14.0/24 is directly connected, Vlan2

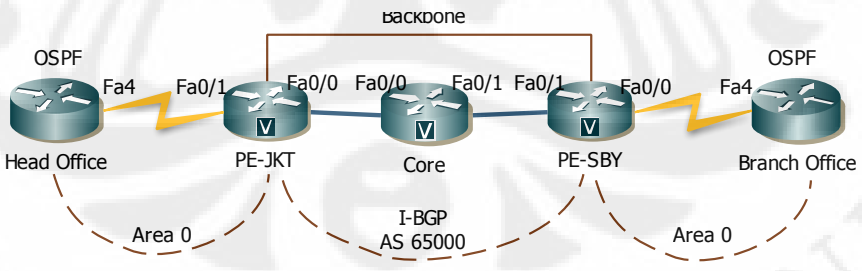
Dari semua pengecekan diatas dapat dipastikan bahwa semua router sudah mendapatkan *routing EIGRP*. Selanjutnyan memastikan *routing EIGRP* untuk *VRF* company-a dapat dilakukan pada salah satu router *PE* seperti dibawah ini :

PE-SBY#sh ip rou vrf company-a
Routing Table: company-a
...
Gateway of last resort is not set
10.0.0.0/8 is variably subnetted, 6 subnets, 3 masks
B 10.10.10.0/24 [200/30720] via 200.20.20.100, 00:17:56
B 10.10.11.0/30 [200/0] via 200.20.20.100, 00:17:56
C 10.10.13.0/30 is directly connected, FastEthernet0/0
B 10.10.12.1/32 [200/156160] via 200.20.20.100, 00:17:56
D 10.10.15.1/32 [90/156160] via 10.10.13.2, 00:19:03, FastEthernet0/0
D 10.10.14.0/24 [90/30720] via 10.10.13.2, 00:19:03, FastEthernet0/0

Dari hasil pengecekan diatas dapat dilihat bahwa company-a mendapatkan *routing EIGRP* dan *MP-BGP*, dimana *MP-BGP* nantinya akan membawa *routing EIGRP* company-a melintasi *MPLS backbone.*

3.3.3 Skenario 3 – *OSPF*

Skenario ketiga adalah implementasi *routing OSPF* pada sistem jaringan yang telah dibuat. Sama halnya dengan skenario 1 dan 2, perbedaan skenario 3 terletak pada aktivasi *routing protocol* baik pada router secara fisik maupun *virtual* dan me-*redistribute*-nya dari dan ke *routing protocol* yang lain, dalam hal ini *BGP*.



Gambar 3.8 Jaringan *VPN* menggunakan *routing protocol OSPF*

3.3.3.1 Implementasi *MPLS Backbone*

Langkah 1 s/d 2 sama dengan skenario pertama. Langkah-langkah untuk mengaktifkan *OSPF* pada *cloud MPLS* antara lain :

- Aktivasi *routing OSPF* di router PE-JKT, *Core* dan PE-SBY

Konfigurasi di router PE-JKT :

```
router OSPF 100
 log-adjacency-changes
 network 200.20.20.0 0.0.0.7 area 0
 network 200.20.20.100 0.0.0.0 area 0
```

Konfigurasi di router *Core* :

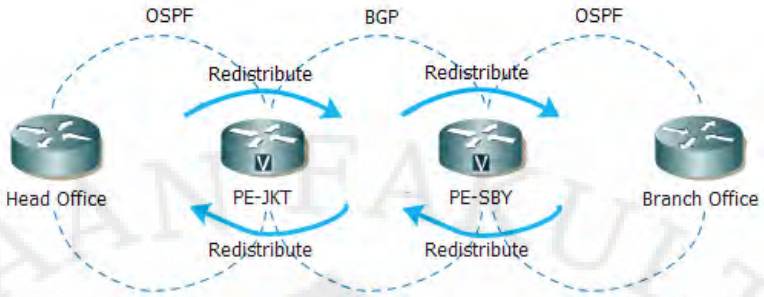
```
router OSPF 100
 log-adjacency-changes
 network 200.20.20.0 0.0.0.7 area 0
 network 200.20.20.101 0.0.0.0 area 0
```

Konfigurasi di router PE-SBY :

```
router OSPF 100
 log-adjacency-changes
 network 200.20.20.0 0.0.0.7 area 0
 network 200.20.20.102 0.0.0.0 area 0
```

OSPF dikonfigurasi dengan perintah **router** *OSPF process-id***.** Process-id adalah nilai antara 1 dan 65535 yang digunakan sebagai identitas suatu proses *routing OSPF* pada router. Nilai process-id bersifat lokal, artinya tidak mempengaruhi router *OSPF* untuk membangun hubungan dengan router tetangganya. Router-router *OSPF* yang berada di dalam *cloud MPLS* dikonfigurasi dengan proses-id 100, berbeda dengan router *OSPF* di router *CE* yang dikonfigurasi dengan *process-id* 101. Sedangkan untuk meng-a*dvertise* IP *network* pada *interface* router yang *directly connected*, konfigurasinya adalah ***network*** *address wildcard-mask* **area** *area-id.* Karena *OSPF* tergolong *classless routing* maka address adalah subnet dari *IP network* dari *interface* tersebut, dan IP Loopback0 perlu di-*advertise* karena *OSPF* akan menjadikan IP Loopback tersebut sebagai router-id, wildcard-mask mewakili *set* alamat *host* yang mendukung segmen, sedangkan area id menetapkan area yang akan disatukan dengan alamat, dimana area 0 didefinisikan sebagai jaringan *backbone* dan pada jaringan ini masing-masing *network* company-a dan *service provider* didefinisikan sebagai *single* area *OSPF*.

- Mengaktifkan *OSPF* pada router *virtual* dan me-*redistribute OSPF* ke *BGP* dan *BGP* kem *OSPF* di router PE-JKT dan PE-SBY



Gambar 3.9 Analogi *redistribute OSPF* ke *BGP* dan *BGP* ke *OSPF*

Redistribute dari *OSPF* ke *BGP*

```
router ospf 100
...
router ospf 101 vrf company-a
 log-adjacency-changes
 redistribute bgp 65000 subnets
 network 10.10.11.0 0.0.0.3 area 0
```

Redistribute dari *BGP* ke *OSPF*

```
router bgp 65000
...
 address-family ipv4 vrf company-a
 redistribute ospf 101
 no auto-summary
 no synchronization
 exit-address-family
```

Konfigurasi diatas menunjukkan *routing* protokol *OSPF* dengan *process-id* 101 (dari router CE) di-*redistribute* ke *BGP* di router PE, dengan tujuan agar *routing* tersebut dapat dibawa oleh *MP-BGP* melintasi *MPLS* backbone dan ketika sampai diujung *PE* yang lain, *BGP* akan di-*redistribute* lagi ke *OSPF*.

3.3.3.2 Implementasi Router *CE*

IP Address yang dikonfigurasi pada setiap *interface* router *CE* sama dengan skenario 1 dan 2, perbedaannya terdapat pada aktivasi *routing OSPF* pada setiap router *CE*. Konfigurasi di router CE adalah sebagai berikut :

Konfigurasi di router *Head Office* :

```
router ospf 101
log-adjacency-changes
network 10.10.11.0 0.0.0.3 area 0
```

network 10.10.10.0 0.0.0.255 area 0
network 10.10.12.0 0.0.0.0 area 0

Konfigurasi di router *Branch Office* :
router ospf 101
log-adjacency-changes
network 10.10.13.0 0.0.0.3 area 0
network 10.10.14.0 0.0.0.255 area 0
network 10.10.15.0 0.0.0.0 area 0

3.3.3.3 Pengecekan *Routing Protocol OSPF*

Untuk memastikan *routing OSPF* apakah sudah aktif disemua router, dapat dilakukan pengecekan seperti dibawah ini :

Head-Office# sh ip rou
CoDES: C - connected, S - static, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2
i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
*ia - IS-IS inter area, * - candidate default, U - per-user static route*
o - ODR, P - periodic downloaded static route
Gateway of last resort is not set
10.0.0.0/8 is variably subnetted, 5 subnets, 3 masks
C 10.10.10.0/24 is directly connected, Vlan2
C 10.10.11.0/30 is directly connected, FastEthernet4
O IA 10.10.13.0/30 [110/2] via 10.10.11.1, 00:06:35, FastEthernet4
C 10.10.12.1/32 is directly connected, Loopback0
O IA 10.10.14.0/24 [110/3] via 10.10.11.1, 00:06:35, FastEthernet4

PE-JKT# sh ip rou
...
Gateway of last resort is not set
200.20.20.0/24 is variably subnetted, 5 subnets, 3 masks
C 200.20.20.100/32 is directly connected, Loopback0
O 200.20.20.101/32 [90/156160] via 200.20.20.2, 00:19:50, FastEthernet0/0
O 200.20.20.102/32 [90/158720] via 200.20.20.2, 00:19:50, FastEthernet0/0
O 200.20.20.8/30 [90/30720] via 200.20.20.2, 00:19:50, FastEthernet0/0
C 200.20.20.0/29 is directly connected, FastEthernet0/0

CORE# sh ip rou
...
Gateway of last resort is not set
200.20.20.0/24 is variably subnetted, 5 subnets, 3 masks
O 200.20.20.100/32 [110/2] via 200.20.20.1, 00:05:56, FastEthernet0/0
C 200.20.20.101/32 is directly connected, Loopback0
O 200.20.20.102/32 [110/2] via 200.20.20.10, 00:05:56, FastEthernet0/1
C 200.20.20.8/30 is directly connected, FastEthernet0/1
C 200.20.20.0/29 is directly connected, FastEthernet0/0

PE-SBY# sh ip rou
...
Gateway of last resort is not set
200.20.20.0/24 is variably subnetted, 5 subnets, 3 masks
O 200.20.20.100/32 [110/3] via 200.20.20.9, 00:05:17, FastEthernet0/1
O 200.20.20.101/32 [110/2] via 200.20.20.9, 00:05:17, FastEthernet0/1

C 200.20.20.102/32 is directly connected, Loopback0
C 200.20.20.8/30 is directly connected, FastEthernet0/1
O 200.20.20.0/29 [110/2] via 200.20.20.9, 00:05:17, FastEthernet0/1

Branch-Office#sh ip rou
...
Gateway of last resort is not set
10.0.0.0/8 is variably subnetted, 6 subnets, 3 masks
O IA 10.10.10.0/24 [110/3] via 10.10.13.1, 00:02:56, FastEthernet4
O IA 10.10.11.0/30 [110/2] via 10.10.13.1, 00:02:56, FastEthernet4
C 10.10.13.0/30 is directly connected, FastEthernet4
O IA 10.10.12.1/32 [110/3] via 10.10.13.1, 00:02:56, FastEthernet4
C 10.10.15.1/32 is directly connected, Loopback0
C 10.10.14.0/24 is directly connected, Vlan2

Dari semua pengecekan diatas dapat dipastikan bahwa semua router sudah mendapatkan *routing OSPF*. Selanjutnyan memastikan *routing OSPF* untuk vrf company-a dapat dilakukan pada salah satu router PE seperti dibawah ini :

PE-JKT#sh ip rou vrf company-a
Routing Table: company-a
...
Gateway of last resort is not set
10.0.0.0/8 is variably subnetted, 5 subnets, 3 masks
O 10.10.10.0/24 [110/2] via 10.10.11.2, 00:06:33, FastEthernet0/1
C 10.10.11.0/30 is directly connected, FastEthernet0/1
B 10.10.13.0/30 [200/0] via 200.20.20.102, 00:06:06
O 10.10.12.1/32 [110/2] via 10.10.11.2, 00:06:33, FastEthernet0/1
B 10.10.14.0/24 [200/2] via 200.20.20.102, 00:06:06

Dari hasil pengecekan diatas dapat dilihat bahwa company-a mendapatkan *routing OSPF* dan *MP-BGP*, dimana *MP-BGP* nantinya akan membawa *routing OSPF* company-a melintasi *MPLS backbone.*

3.4 Hipotesa

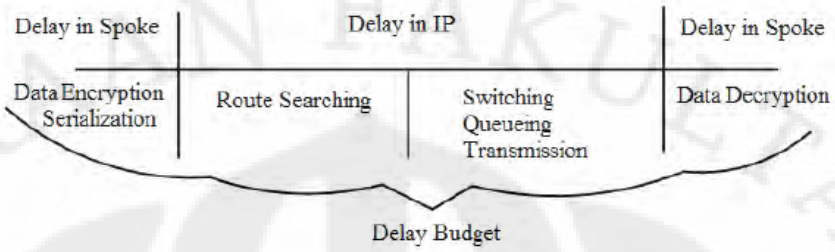
Ruta Jankuniene dan Ieva Jankuinaite dalam papernya melakukan uji coba mengenai pengaruh pembentukan/pemilihan rute pada *routing* protokol RIP, *EIGRP* dan *OSPF* terhadap *QoS* pada jaringan *DMVPN* (solusi *VPN* standar Cisco Corporation).

Tabel 3.3 *Routing protocol classification* [5]

Protocol	Type	Routing algorithm	Open standard	Network type	Route control	Convergence	CPU	Scalability
EIGRP	IGP	DV	No	hub-to-spoke spoke-to-spoke	Good	Faster	High	Lower
OSPF	IGP	LS	Yes	hub-to-spoke spoke-to-spoke	Medium	Faster	High	Lower
RIP	IGP	DV	Yes	hub-to-spoke*	Poor	Lower	Low	High

Note: * may be used for creating spoke-to-spoke tunnel.

Dari paper tersebut dinyatakan bahwa salah satu indikator kualitas yang paling penting untuk layanan *real-time* adalah delay paket, dimana besarnya *delay* didefinisikan pada Gambar 3.10.



Gambar 3.10 *Packet flow* delay budget [5]

Dari paper tersebut disimpulkan bahwa *EIGRP* adalah *routing protocol* yang paling efektif daripada *RIPv2* dan *OSPF* untuk diimplementasikan pada jaringan *DMVPN*.

Dalam tugas akhir ini akan dilakukan ujicoba pengaruh penggunaan *routing* protokol *RIPv2*, *EIGRP* dan *OSPF* pada jaringan *MPLS* terhadap *tunnel IPSec* untuk layanan *video conference* yang bersifat *real-time*. Mengacu pada paper tersebut, diharapkan dapat diketahui *routing protocol* yang paling optimal untuk diimplementasikan pada jaringan *MPLS* dengan tunnel *IPSec* pada skala *network* yang kecil dan *bandwidth* yang terbatas.

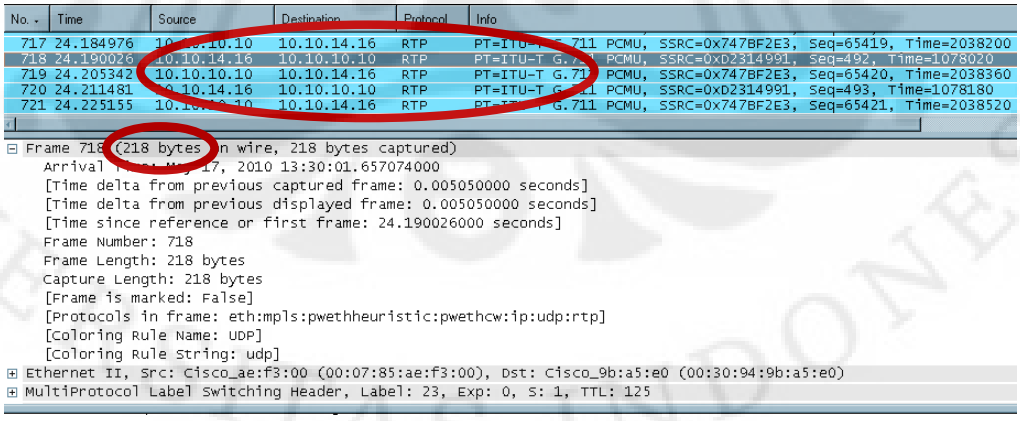
BAB 4
PENGUJIAN DAN ANALISA SISTEM

4.1 Pengujian Sistem

Pada pengujian sistem ini akan dianalisa pengaruh penentuan/pemilihan rute oleh *routing protocol* yang digunakan terhadap *QoS* sebelum dan sesudah diimplementasikan *IPSec* pada jaringan *MPLS* untuk aplikasi *video conference*. Parameter yang akan dianalisa meliputi *delay, jitter, packet loss* dan *throughput* dengan bantuan wireshark.

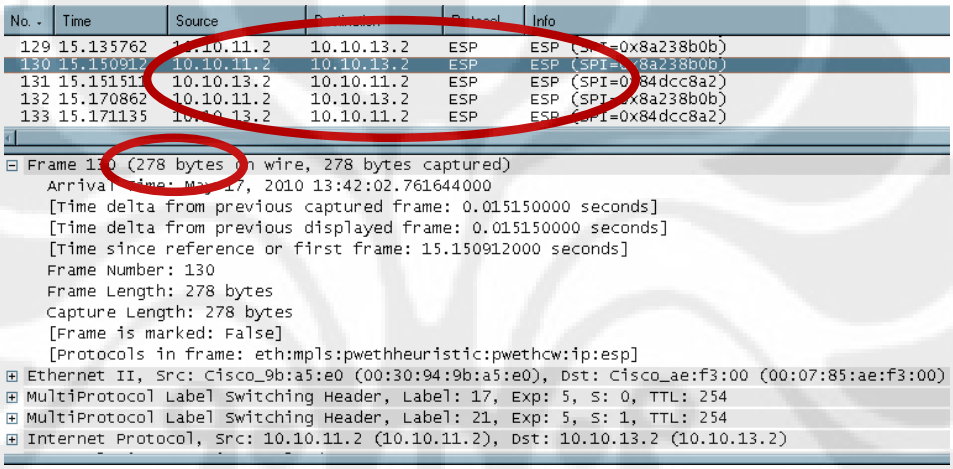
4.1.1 Pengujian *IPSec*

IPSec diimplementasikan pada setiap router *CE* dengan tujuan agar koneksi *end-to-end* router menjadi *secure*, karena paket data akan dienkripsi terlebih dahulu sebelum dikirim dan didekripsi sebelum diterima. Hal ini dapat diketahui dengan melihat paket data yang melintasi sepanjang router *Head Office* hingga router *Branch Office*. Untuk memastikan hal tersebut, digunakan *software* wireshark yang di-install pada *PC* yang dihubungkan ke salah satu port ethernet switch, dimana switch tersebut dipasang pada koneksi antara router *Head Office* dan PE-JKT dengan tujuan untuk menangkap paket data yang melintasi rute tesebut dan memastikan apakah paket data yang lewat sudah benar-benar ter-enkripsi atau belum. Hasil penyadapan dengan wireshark diperoleh seperti Gambar 4.1 dan 4.2.



Gambar 4.1 Paket data sebelum implementasi *IPSec*

Paket *RTP* (*Real Time Protocol*) dari *video conference* yang dilakukan dapat ditangkap oleh wireshark sekaligus dengan *codec* audio dan video yang digunakan, yaitu G.711 untuk audio dan H.263 untuk video, paket *RTP* inilah yang nantinya akan dianalisa untuk mengetahui pengaruh algoritma *routing* yang digunakan terhadap implementasi *IPSec* pada jaringan *MPLS*. Dari Gambar 4.1 dapat diketahui bahwa dengan jaringan *MPLS* saja tidak cukup aman untuk melakukan komunikasi data seperti *video conference,* karena paket data yang melintasi jaringan masih dapat disadap dan dilihat isi paketnya, sehingga masih memungkinkan terjadinya akses ilegal oleh pihak lain.



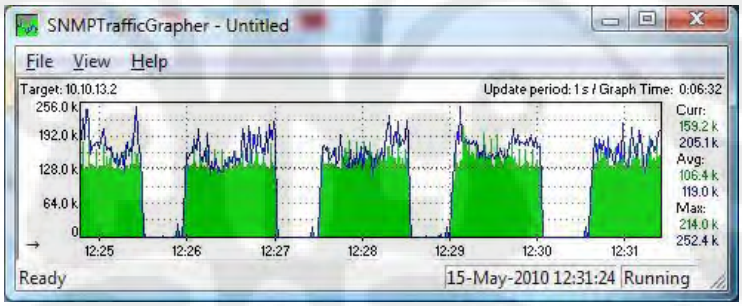
Gambar 4.2 Paket data setelah implementasi *IPSec*

Dapat dilihat pada Gambar 4.2 bahwa semua paket *RTP* yang melintasi jaringan tertangkap oleh wireshark sebagai paket *ESP* (*Encapsulation Security Payload*) setelah diiimplementasikan *IPSec*, dimana *ESP* tersebut merupakan protokol *security* yang telah diimplementasikan pada *end-to-end* router sebagaimana yang sudah dijelaskan pada Bab 3. Selain itu dapat dilihat dari informasi yang tersedia pada wireshark bahwa rata-rata ukuran frame setelah implementasi *IPSec* lebih besar daripada sebelum implementasi *IPSec*. Hal ini disebabkan oleh adanya penambahan *header-header ESP* pada data yang dikirimkan ketika berada di router *CE.* Selain itu, *IP addre source* dan *destination* yang terdeteksi merupakan *IP gateway* dari masing-masing router, sehingga tidak dapat diketahui host yang melakukan *video conference.* Dengan demikian dapat dipastikan bahwa trafik data yang melintasi router *Head Office* hingga router *Branch Office* sudah terjamin keamanannya, karena paket data yang

sesungguhnya tidak dapat dilihat oleh pihak ketiga yang tidak mempunyai akses didalamnya.

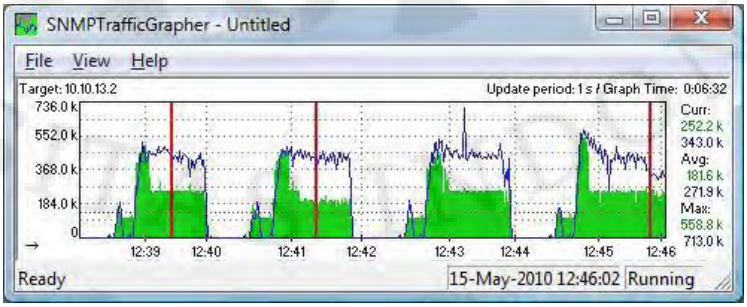
4.1.2 Penentuan *Bandwidth* Minimum

Sebelum merencanakan trafik video di jaringan, perlu dipastikan bahwa *bandwidth* cukup untuk semua aplikasi yang dibutuhkan. Dimulai dengan menghitung kebutuhan *minimum bandwidth* untuk setiap aplikasi utama, dalam hal ini audio dan video. Jumlahnya akan menunjukkan kebutuhan *minimum bandwidth* yang diperlukan untuk satu *link* dan harus mengkonsumsi tidak lebih dari 75% dari total *bandwidth* yang tersedia di satu *link*. Aturan 75% muncul dengan asumsi bahwa sejumlah *bandwidth* dibutuhkan untuk trafik *overhead* dan aplikasi-aplikasi tambahan yang lain. Dari grafik pada Gambar 4.3 terlihat bahwa rata-rata *bandwidth* minimum yang diperlukan untuk melewatkan trafik suara dan video adalah 192 Kbps, maka dengan asumsi diatas, total *bandwidth* yang diimplementasikan pada jaringan ini adalah 256 Kbps.



Gambar 4.3 Alokasi *bandwidth* untuk *video conference* 2 *user*

Pada saat dilakukan *video conference* dengan 2 user, trafik data yang melintasi jaringan masih normal/stabil. Pada Gambar 4.4 terlihat bahwa trafik meningkat dengan tajam pada saat user yang melakukan *video conference* bertambah 1, bahkan jaringan menjadi *overload* dan koneksi sempat terputus.



Gambar 4.4 Alokasi *bandwidth* untuk *video conference* 3 *user*

Pada saat kondisi jaringan yang tidak stabil ini, performansi masing-masing *routing protocol* akan kembali diukur dan dilihat pengaruhnya terhadap kualitas *video conference* yang dilakukan.

4.2 Sistematika Pengukuran

Parameter yang akan diukur dengan wireshark meliputi : *delay, jitter, packet loss* dan *throughput*. Parameter-parameter tersebut akan dianalisa dengan membandingkan nilai yang terukur pada wireshark pada setiap skenario, dimana telah dijelaskan di awal bahwa pada setiap skenario diimplementasikan *routing* protokol yang berbeda, tujuannya adalah untuk mengetahui pengaruh algoritma *routing* terhadap *QoS* sebelum dan sesudah implementasi *IPSec* pada *end-to-end* router dalam sebuah jaringan *MPLS* dan mengetahui *routing protocol* yang paling optimal untuk jaringan tersebut.

Untuk memperkuat hasil pengukuran pada masing-masing skenario, data diambil pada dua kondisi yang berbeda, yaitu pada saat jaringan stabil, artinya trafik dalam keadaan normal dan pada saat jaringan tidak stabil yang terjadi pada saat *video conference* 3 *user* yang menyebabkan trafik penuh/*overload*. Oleh karena itu untuk mencapai kedua kondisi tersebut diperlukan *bandwidth* yang harus diimplementasikan pada jaringan yang telah dibangun.

Pengukuran *delay, jitter, packet loss* dan *throughtput* dilakukan melalui *call setup* oleh *user* 2 yang berada di *Branch Office* ke *user* 1 yang berada di *Head Office*, *PC* dengan wireshark dihubungkan ke interface router *Branch Office*, satu *LAN* dengan *user* 2 untuk menangkap paket *RTP* dari *video conference* yang dilakukan dan *user* 3 , dimana *video conference* tersebut masing-masing dilakukan selama 1 menit untuk kemudian diukur dan dianalisa parameter-parameternya.

4.3 Data Hasil Pengukuran

Pengukuran parameter *QoS* didasarkan pada codec yang digunakan, yaitu G.711 untuk audio dan H.263 untuk video. Dari 3 skenario yang telah diimplementasikan dan beberapa kali percobaan, diperoleh hasil pengambilan data rata-rata dari setiap *routing* protokol yang digunakan. Pada kondisi jaringan stabil, data yang akan dianalisa dapat dilihat pada Tabel 4.1 untuk jaringan sebelum implementasi *IPSec* dan Tabel 4.2 untuk jaringan setelah implementasi *IPSec*.

Tabel 4.1 Nilai parameter *QoS video conference* 2 user sebelum implementasi *IPSec*

Parameter	*RIPv2*		*EIGRP*		*OSPF*	
	G.711	**H.263**	**G.711**	**H.263**	**G.711**	**H.263**
Delay (ms)	19.99	137.40	19.99	77.01	19.99	77.07
Jitter (ms)	1.02	9.55	0.91	26.93	0.84	27.01
Packet loss (%)	0.00	0.00	0.00	0.00	0.00	0.00
Throughput (packet/s)	50.01	7.29	50.02	12.98	50.01	12.97

Tabel 4.2 Nilai parameter *QoS video conference* 2 user setelah implementasi *IPSec*

Parameter	*RIPv2*		*EIGRP*		*OSPF*	
	G.711	**H.263**	**G.711**	**H.263**	**G.711**	**H.263**
Delay (ms)	19.99	147.73	19.99	83.46	20.25	154.71
Jitter (ms)	1.03	9.72	1.08	25.14	2.30	11.79
Packet loss (%)	0.00	0.00	0.00	0.00	0.00	0.00
Throughput (packet/s)	50.02	6.78	50.02	12.21	49.41	6.47

Sedangkan pada kondisi jaringan tidak stabil, yaitu pada saat melakukan *video conference* dengan 3 user, data rata-rata yang telah diperoleh dan akan dianalisa dapat dilihat pada Tabel 4.3 untuk jaringan sebelum implementasi *IPSec* dan tabel 4.4 untuk jaringan setelah implementasi *IPSec*.

Tabel 4.3 Nilai parameter QoS *video conference* 3 user sebelum implementasi *IPSec*

Parameter	*RIPv2*		*EIGRP*		*OSPF*	
	G.711	**H.263**	**G.711**	**H.263**	**G.711**	**H.263**
Delay (ms)	22.17	148.48	23.10	122.85	22.41	110.68
Jitter (ms)	1.89	25.07	2.76	25.34	1.27	25.77
Packet loss (%)	3.20	38.93	4.78	27.42	3.04	18.67
Throughput (packet/s)	46.59	11.69	45.67	11.33	46.26	11.26

Tabel 4.4 Nilai parameter QoS *video conference* 3 user setelah implementasi *IPSec*

Parameter	*RIPv2*		*EIGRP*		*OSPF*	
	G.711	**H.263**	**G.711**	**H.263**	**G.711**	**H.263**
Delay (ms)	25.44	286.34	27.06	215.55	24.94	381.08
Jitter (ms)	3.82	11.49	1.89	24.86	2.08	8.58
Packet loss (%)	5.07	35.18	6.47	54.80	4.60	51.74
Throughput (packet/s)	41.46	5.81	39.66	11.37	42.05	6.22

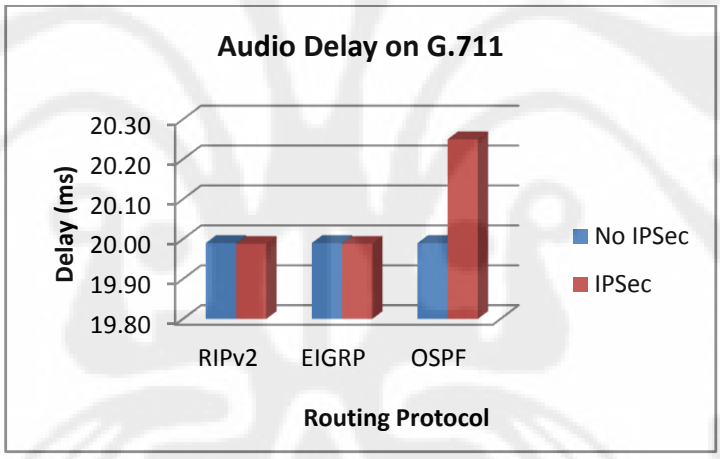
4.4 Analisa Data Hasil Pengukuran

4.4.1 Analisa Hasil Pengukuran *Delay*

Delay atau *latency* merupakan waktu yang diperlukan oleh paket data dari terminal pengirim hingga mencapai terminal penerima. *Delay* merupakan salah satu permasalahan yang harus diperhitungkan karena kualitas suara dan gambar menjadi baik sangat tergantung dari delay yang terjadi. Besarnya *delay* maksimum yang direkomendasikan oleh *ITU-T* pada G.114 untuk aplikasi suara dan gambar adalah 150 ms, sedangkan *delay* maksimum dengan kualitas suara dan gambar yang masih dapat diterima pengguna adalah 250 ms.

4.4.1.1 Kondisi Jaringan Stabil

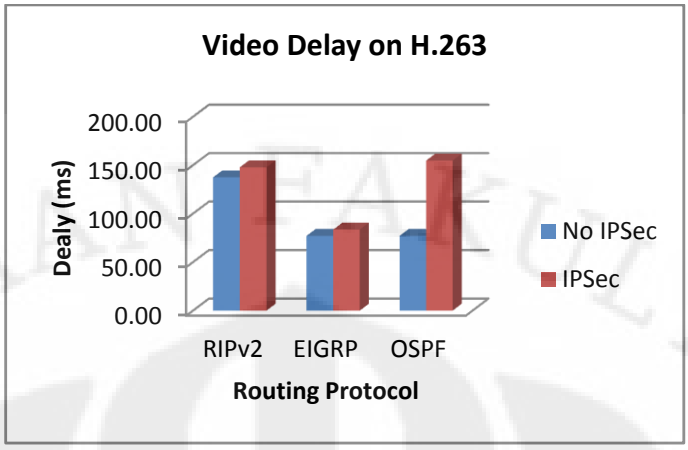
Kondisi ini terjadi pada saat melakukan *video conference* 2 *user*, dimana *bandwidth* masih mencukupi dan tidak terjadi putus koneksi.



Gambar 4.5 *Delay* rata-rata G.711 pada kondisi jaringan stabil

Pada Gambar 4.5, *delay* audio pada *codec* G.711 baik untuk *RIPv2*, *EIGRP* dan *OSPF* relatif sama sebelum dan setelah diimplementasikan *IPSec*, yaitu rata-rata 20 ms, delay *OSPF* relatif lebih besar setelah implementasi IPSec. Hal ini disebabkan oleh ukuran frame audio dengan *codec* G.711 relatif kecil, yaitu 0.125 ms, sehingga implementasi *IPSec* yang menambah ukuran paket dengan adanya proses enkapsulasi (penambahan *header ESP*, dll) tidak berpengaruh banyak terhadap *delay* audio.

Namun lain halnya dengan *delay* video dengan *codec* H.263 yang terlihat pada Gambar 4.6, dimana *delay* mengalami kenaikan setelah diimplementasikan *IPSec*.



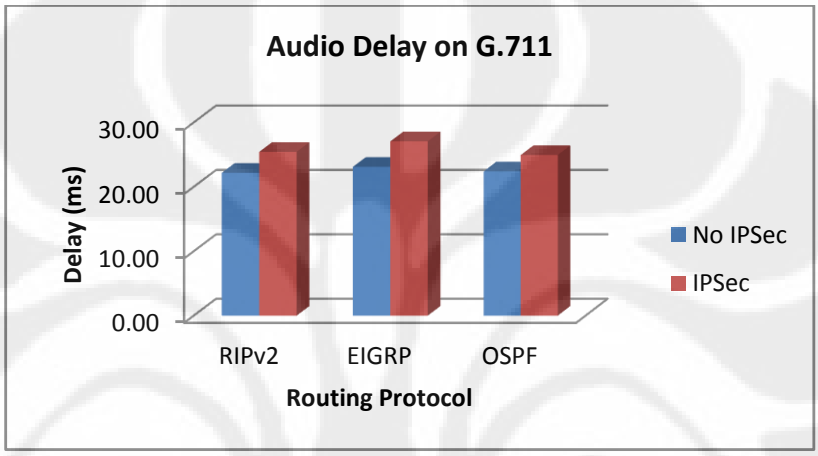
Gambar 4.6 *Delay* rata-rata G.711 pada kondisi jaringan stabil

Dalam kondisi jaringan stabil, performansi *video conference* untuk semua *routing* protokol tergolong bagus, meskipun untuk *RIPv2* mempunyai *delay* video yang paling besar bila dibandingkan dengan *EIGRP* dan *OSPF*, yaitu sekitar 137 ms. Hal ini disebabkan oleh *RIPv2* menggunakan algoritma *distance vector* yang mengkonsumsi lebih banyak *bandwidth* karena pengiriman update tabel *routing* yang dilakukan secara periodik, yaitu setiap 30 detik. Sedangkan untuk *EIGRP* dan *OSPF* mempunyai *delay* yang relatif sama dan lebih kecil daripada *RIPv2* yaitu sekitar 77 ms, karena kedua *routing protocol* tersebut hanya melakukan update *routing* pada saat terjadi perubahan dalam jaringan. Setelah diimplemetasikan *IPSec* terlihat jelas bahwa masing-masing *routing* protokol *RIPv2*, *EIGRP* dan *OSPF* mengalami kenaikan *delay*, karena pada dasarnya implementasi *IPSec* akan menambah proses pada router dan menambah ukuran paket data. Kenaikan *delay* yang terjadi pada *RIPv2* dan *EIGRP* setelah implementasi *IPSec* tidak terjadi secara signifikan yaitu sekitar 8%, lain halnya dengan *OSPF* yang mengalami kenaikan *delay* secara signifikan yaitu sekitar 101%, hal ini disebabkan oleh beban kerja router *OSPF* semakin bertambah karena selain perhitungan *cost* yang lebih kompleks daripada *EIGRP* dan *RIPv2* untuk mencari rute terbaik, terdapat *delay budget* lain yang diutarakan oleh Ruta Jankuniene dan Ieva Jankuinaite akibat implementasi *IPSec* yaitu delay enkripsi, *switching, queueing, transmission* dan dekripsi, adanya *delay budget* tersebut semakin menambah proses pada router sehingga *delay* mengalami kenaikan secara signifikan. Sedangkan untuk *RIPv2*, implementasi IPSec tidak banyak berpengaruh terhadap kenaikan *delay* selama tidak terjadi kegagalan jaringan.

Demikian juga dengan *EIGRP*, kenaikan *delay* yang terjadi relatif lebih kecil daripada *OSPF* dikarenakan perhitungan *metric EIGRP* tidak banyak menambah beban router, sehingga delay budget akibat implementasi *IPSec* relatif lebih kecil.

4.4.1.2 Kondisi Jaringan Tidak Stabil

Kondisi ini terjadi pada saat dilakukan *video conference* dengan 3 *user*, dimana trafik data menjadi penuh/*overload* yang menyebabkan koneksi menjadi tidak stabil, *delay* yang terukur pada kondisi ini dapat dilihat pada Gambar 4.7.

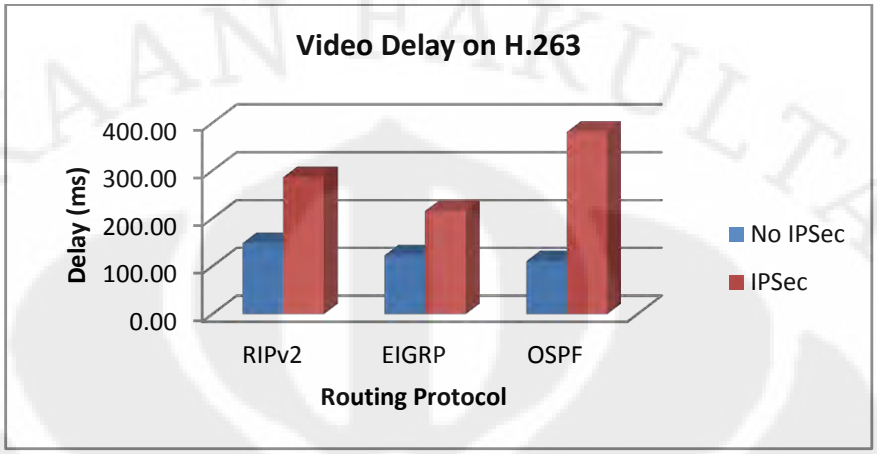


Gambar 4.7 *Delay* rata-rata G.711 pada kondisi jaringan tidak stabil

Pada kondisi jaringan tidak stabil, *delay* audio untuk *RIPv2*, *EIGRP* dan *OSPF* setelah implementasi *IPSec* mengalami kenaikan rata-rata sekitar 14%, *delay EIGRP* relatif lebih besar. Dari grafik yang ditunjukkan oleh Gambar 4.5 dan 4.7 terlihat bahwa adanya implementasi *IPSec* tidak banyak berpengaruh terhadap *delay* audio.

Pada saat terjadi kegagalan jaringan akibat dari trafik *overload* hingga putusnya koneksi, *RIPv2* tidak terlalu menambah beban pada jaringan, tetapi membutuhkan waktu lama untuk mencapai konvergen, sehingga *delay* video yang dihasilkan pun mengalami kenaikan relatif tinggi, yaitu sekitar 93% . Sedangkan *EIGRP* memiliki kenaikan *delay* yang relatif kecil setelah implementasi *IPSec* dibandingkan dengan *RIPv2* dan *OSPF*, yaitu sekitar 75%. Hal ini disebabkan oleh *EIGRP* memiliki tabel topologi yang juga menyimpan rute cadangan (*feasible successor)* selain rute utama/terbaik, sehingga jika rute utama mengalami kegagalan maka *EIGRP* dapat menggunakan rute cadangan tersebut untuk membangun koneksi kembali. Jika jaringan tidak mempunyai rute cadangan,

maka router *EIGRP* akan dengan cepat meminta router-router tetangganya untuk mencarikan rute baru dan memanfaatkan informasi dari router tetangganya tersebut untuk membangun rute kembali.



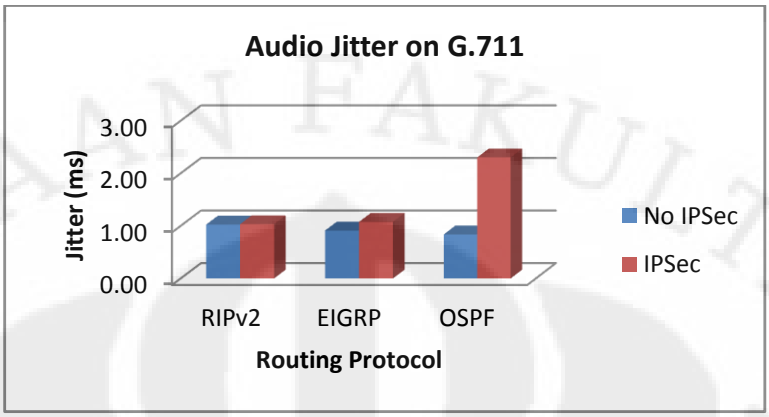
Gambar 4.8 *Delay* rata-rata H.263 pada kondisi jaringan tidak stabil

Pada kondisi jaringan stabil, *OSPF* dapat menghemat penggunaan *bandwidth* seperti halnya *EIGRP*, tetapi ketika jaringan tidak stabil dan koneksi terputus, maka *OSPF* akan mengirimkan paket *LSA* secara *flooding*, sehingga jaringan dibanjiri oleh paket-paket *LSA* dan menyebabkan beban jaringan bertambah, akibatnya *delay* mengalami kenaikan secara signifikan setelah implementasi *IPSec*, yaitu sebesar 244%. Kenaikan delay pada masing-masing *routing* protokol membuat kualitas *video conference* menurun, kualitas suara menurun, kadang terdengar putus-putus, tetapi tidak sebesar penurunan kualitas pada gambar, dimana gambar yang terlihat menjadi patah-patah dan kadang tidak jelas, bahkan pada *OSPF*, beberapa kali koneksi sempat terputus dan *video conference* gagal dilakukan karena *delay* terlalu besar.

4.4.2 Analisa Hasil Pengukuran *Jitter*

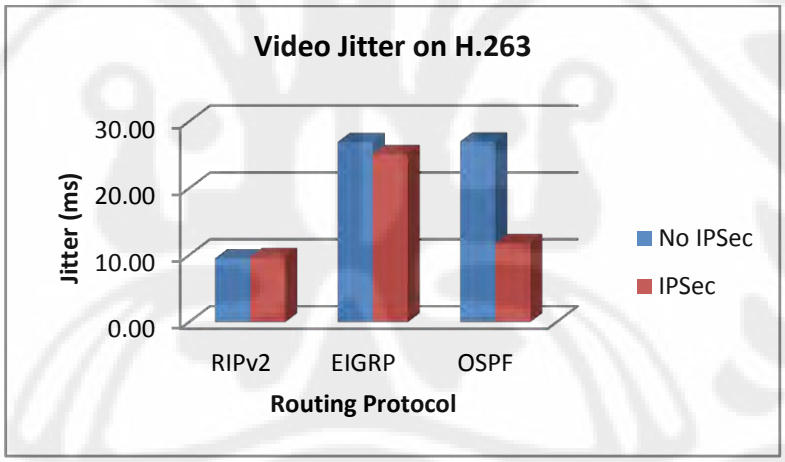
Jitter merupakan variasi *delay* yang terjadi akibat adanya selisih waktu atau *interval* antar kedatangan paket di penerima. Secara sederhana bisa dikatakan bahwa *jitter* adalah perbedaan waktu kedatangan antara satu paket dengan paket berikutnya. Parameter *jitter* perlu dianalisa untuk mengetahui *delay* kedatangan antara satu paket dengan paket lainnya. Sesuai dengan rekomendasi *ITU-T*, bahwa *jitter* yang masih dapat ditoleransi adalah kurang dari 30 ms.

4.4.2.1 Kondisi Jaringan Stabil



Gambar 4.9 *Jitter* rata-rata G.711 pada kondisi jaringan stabil

Pada kondisi jaringan stabil, besarnya *jitter* pada setiap *routing protocol* untuk audio relatif sama, yaitu rata-rata 1 ms. Setelah implementasi *IPSec*, besarnya jitter naik rata-rata menjadi 2 ms. Seperti halnya *delay, jitter* tidak berpengaruh banyak terhadap audio yang menggunakan *codec* G.711 ini.



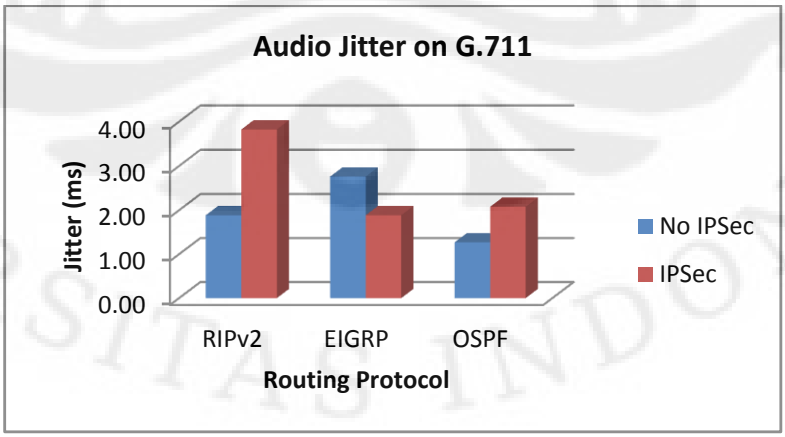
Gambar 4.10 *Jitter* rata-rata H.263 pada kondisi jaringan stabil

Pengaruh *jitter* juga baru dapat dilihat pada trafik video, hal ini terlihat pada Gambar 4.10, dimana *jitter* untuk video pada *EIGRP* dan *OSPF* sebelum implementasi *IPSec* relatif besar, yaitu rata-rata 27 ms, sedangkan *jitter* pada *RIPv2* relatif kecil baik sebelum maupun sesudah implementasi *IPSec*, yaitu rata-rata 9 ms. *Jitter OSPF* mengalami penurunan yang signifikan setelah implementasi *IPSec*, yaitu sekitar 56%, sedangkan pada *EIGRP* hanya mengalami penurunan sebesar 7%.

Namun demikian pengaruh *jitter* pada kinerja jaringan harus dilihat bersama *delay*, dimana ketika *jitter* besar namun *delay* kecil maka kinerja jaringan tidak bisa dikatakan jelek, karena besarnya *jitter* dapat dikompensasi dengan nilai *delay* yang kecil. *Jitter* akan menurunkan kinerja jaringan jika nilainya besar dan *delay* juga besar. Pada *EIGRP*, nilai *jitter* relatif besar daripada *RIPv2* dan *OSPF* , tetapi pada pengukuran *delay* diperoleh hasil yang lebih kecil dari *RIPv2* dan *OSPF*. Pada *RIPv2 jitter* yang dihasilkan relatif kecil, tetapi pada pengukuran *delay* diperoleh nilai yang relatif besar baik sebelum maupun setelah implementasi *IPSec*. Sedangkan pada *OSPF* sebelum implementasi *IPSec* nilai *jitter* relatif besar dan *delay* yang terukur relatif kecil (relatif sama dengan *EIGRP*), tetapi setelah implementasi *IPSec delay* yang terukur menjadi lebih besar dari *EIGRP* dan nilai *jitter* menurun. Sehingga diperoleh hubungan antara *delay* dan *jitter* untuk trafik video, semakin besar *delay* maka *jitter*-nya semakin kecil. Namun merujuk pada rekomendasi *ITU-T*, bahwa *jitter* yang masih dapat ditoleransi adalah kurang dari 30 ms. Ini artinya *jitter* yang dihasilkan pada *RIPv2*, *EIGRP* dan *OSPF* masih dapat ditoleransi sebelum dan sesudah implementasi *IPSec*.

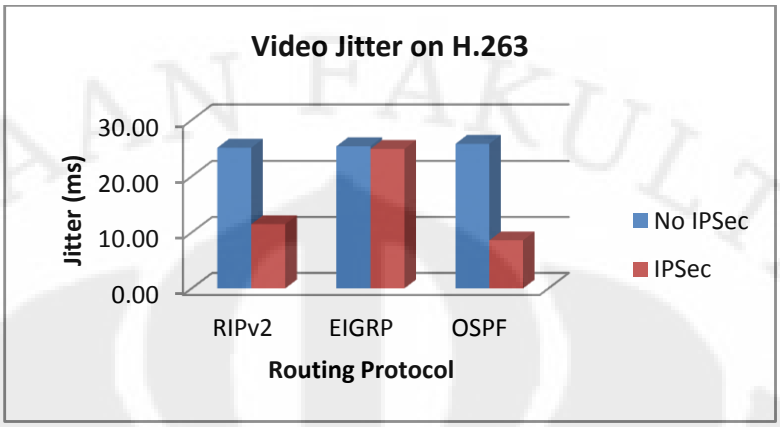
4.4.2.2 Kondisi Jaringan Tidak Stabil

Dari grafik pada Gambar 4.11 terlihat bahwa pada kondisi jaringan tidak stabil, *jitter* untuk audio pun relatif masih kecil, yaitu rata-rata 2 ms dan *jitter EIGRP* relatif lebih besar. Terdapat kenaikan *jitter* setelah implementasi *IPSec* tetapi tidak signifikan.



Gambar 4.11 *Jitter* rata-rata G.711 pada kondisi jaringan tidak stabil

Dari pengukuran *delay* dan *jitter* diatas, diperoleh hubungan semakin besar *delay* maka *jitter* juga semakin besar.



Gambar 4.12 *Jitter* rata-rata H.263 pada kondisi jaringan tidak stabil

Jitter dipengaruhi oleh variasi beban trafik dan besarnya tumbukan antar paket (*congestion*) yang ada dalam jaringan. Pada kondisi jaringan tidak stabil, nilai *jitter* pada *RIPv2*, *EIGRP* dan *OSPF* relatif sama sebelum implementasi *IPSec*, yaitu rata-rata 25 ms, namun setelah implementasi *IPSec jitter* pada *RIPv2* dan *OSPF* mengalami penurunan secara signifikan sebesar 54% dan 67%, sedangkan *jitter EIGRP* hanya mengalami penurunan sebesar 2%. Untuk nilai *jitter* pada kondisi tidak stabil baik sebelum dan setelah implementasi *IPSec* masih dapat ditoleransi karena masih dibawah standar *ITU-T*.

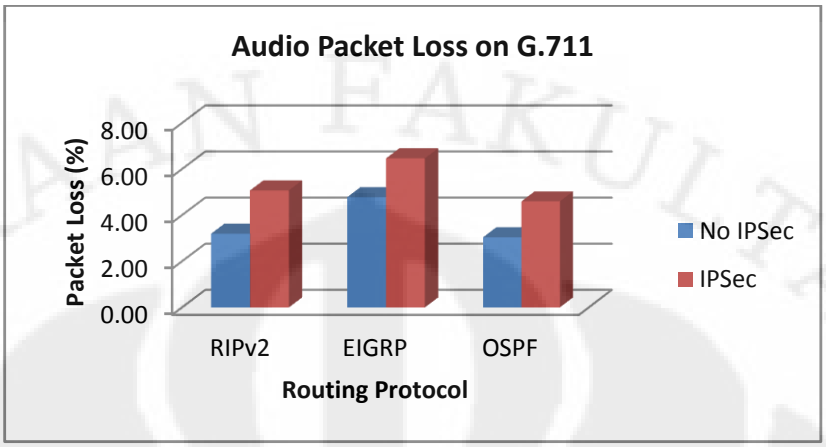
4.4.3 Analisa Hasil Pengukuran *Packet loss*

Packet loss merupakan banyaknya paket yang gagal mencapai tempat tujuan dimana paket tersebut terkirim. Ketika *packet loss* besar maka dapat diketahui bahwa jaringan sedang sibuk atau terjadi *overload*. Sesuai dengan standar *ITU-T,* nilai *packet loss* yang masih dapat ditoleransi adalah 10% s/d 30%.

4.4.3.1 Kondisi Jaringan Stabil

Pada kondisi jaringan stabil, dimana tidak terdapat gangguan pada jaringan (misal: koneksi *down*) dan *bandwidth* masih tersedia, maka pada masing-masing *routing* protokol tidak dijumpai adanya *packet loss* (*packet loss* = 0%) baik pada audio maupun video, baik sebelum maupun setelah implementasi *IPSec*.

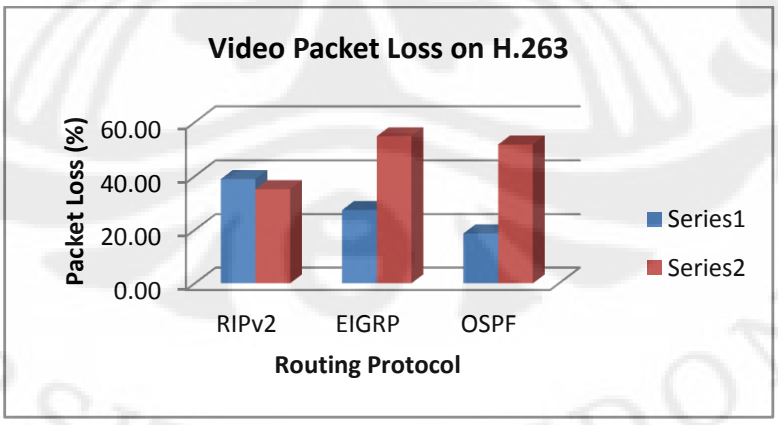
4.4.3.2. Kondisi Jaringan Tidak Stabil



Gambar 4.13 *Packet loss* rata-rata G.711 pada kondisi jaringan tidak stabil

Kondisi jaringan yang tidak stabil bisa diakibatkan oleh banyaknya paket data yang memenuhi atau melebihi kapasitas jaringan (*overload*) hingga putusnya koneksi. Pada saat paket data melebihi kapasitas jaringan, maka akan banyak juga paket data yang hilang. Besarnya *packet loss* yang terjadi sebelum dan sesudah implementasi *IPSec* dapat dilihat pada Gambar 4.13 untuk audio dan 4.14 untuk video.

Rata-rata nilai *packet loss* untuk audio pada *RIPv2*, *EIGRP* dan *OSPF* adalah 3% sebelum implementasi *IPSec* dan nilai *packet loss* pada *EIGRP* relatif lebih besar.



Gambar 4.14 *Packet loss* rata-rata H.263 pada kondisi jaringan tidak stabil

Gambar 4.14, menunjukkan bahwa *EIGRP* dan OSPF memiliki nilai rata-rata *packet loss* yang relatif sama untuk trafik video, yaitu 53%, dimana *packet*

loss EIGRP relatif lebih besar dibandingkan dengan *RIPv2* dan *OSPF*. *Packet loss* yang terjadi juga dipengaruhi oleh besarnya delay, dimana semakin besar delay maka packet lossnya juga semakin besar.

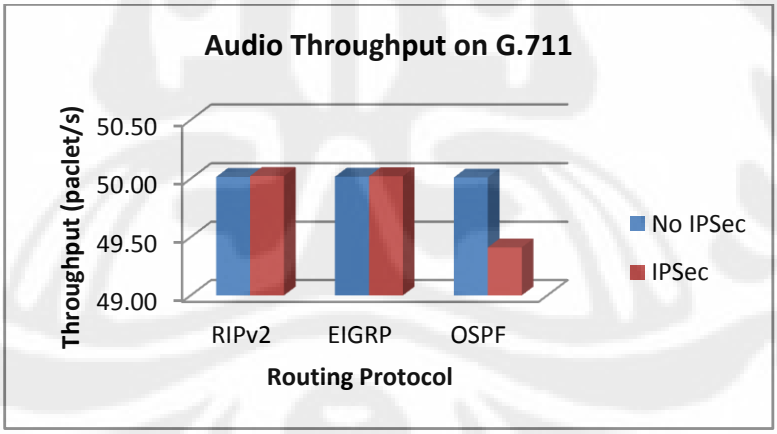
Merujuk pada rekomendasi ITU-T bahwa *packet loss* yang masih dapat ditoleransi adalah sebesar 10% hingga 30%. Dengan demikian *packet loss* yang terjadi pada trafik video untuk *RIPv2*, *EIGRP* dan *OSPF* setelah implementasi *IPSec* sudah tidak memenuhi standar ITU-T.

4.4.4 Pengukuran *Throughput*

Throughput adalah banyaknya paket data yang diterima oleh sebuah *node* dalam selang waktu pengamatan tertentu. Nilai *throughput* dipengaruhi oleh *delay, jitter*, dan *packet loss* yang terjadi.

4.4.3.1 Kondisi Jaringan Stabil

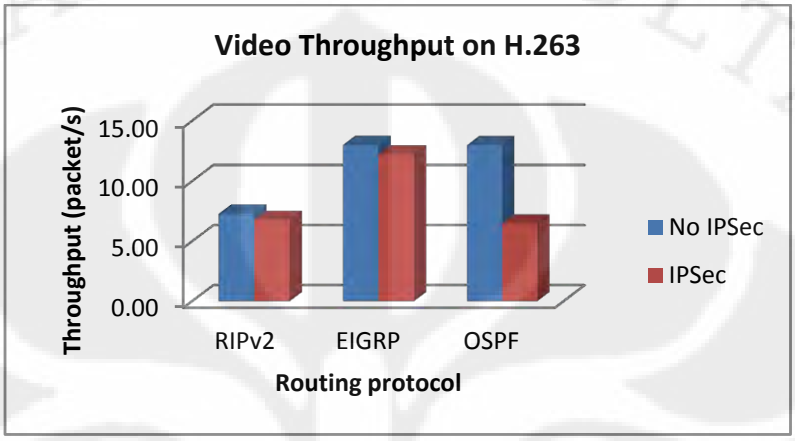
Pada saat jaringan stabil dan tidak terdapat *packet loss*, terlihat pada grafik yang ditunjukkan oleh Gambar 4.15 bahwa nilai *throughput* untuk audio sebelum implementasi *IPSec* pada *RIPv2*, *EIGRP* dan *OSPF* relatif sama yaitu sekitar 50 paket/detik, artinya terdapat 50 paket data yang dapat dikirimkan tiap detiknya.



Gambar 4.15 *Throughput* rata-rata G.711 pada kondisi jaringan stabil

Namun demikian setelah implementasi *IPSec throughput* pada *OSPF* menurun secara signifikan sebesar 50%, sedangkan untuk *RIPv2* dan *EIGRP* hanya menurun sekitar 1% saja. Dilihat dari parameter yang telah diukur, *jitter* pada *OSPF* setelah implementasi *IPSec* mengalami kenaikan secara signifikan yaitu sebesar 175% meskipun masih tergolong cukup bagus. Dari hasil pengukuran ini dapat diketahui bahwa kenaikan *jitter* ternyata bisa berpengaruh terhadap

turunnya *throughput*, karena semakin besar *jitter* pada kondisi stabil memungkinkan terjadinya *congestion* antar paket data meskipun tidak sampai terjadi *packet loss,* sehingga menyebabkan terjadinya peningkatan perbedaan *delay* antar paket, dan akibatnya jumlah paket data yang dikirim pun dapat mengalami penurunan.

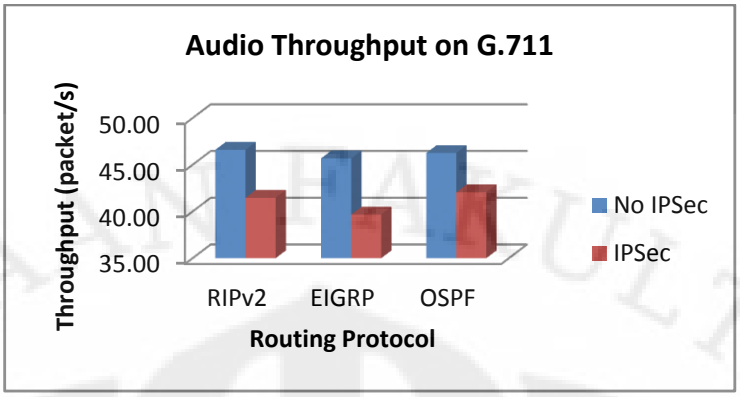


Gambar 4.16 *Throughput* rata-rata H.263 pada kondisi jaringan stabil

Pada kondisi stabil, *throughput* untuk video mempunyai karakteristik semakin besar *jitter* nilai *throughput*-nya akan semakin kecil. Hal ini dapat dilihat pada grafik yang ditunjukkan oleh Gambar 4.16, dimana sebelum implementasi *IPSec* throughput *EIGRP* dan *OSPF* relatif sama, yaitu rata-rata 12 ms, sedangkan throughput *RIPv2* relatif lebih kecil, yaitu rata-rata 7 ms. Namun setelah implementansi *IPSec* throughput *OSPF* menurun signifikan sebesar 50%, sedangkan throughput *EIGRP* dan *RIPv2* relatif tetap.

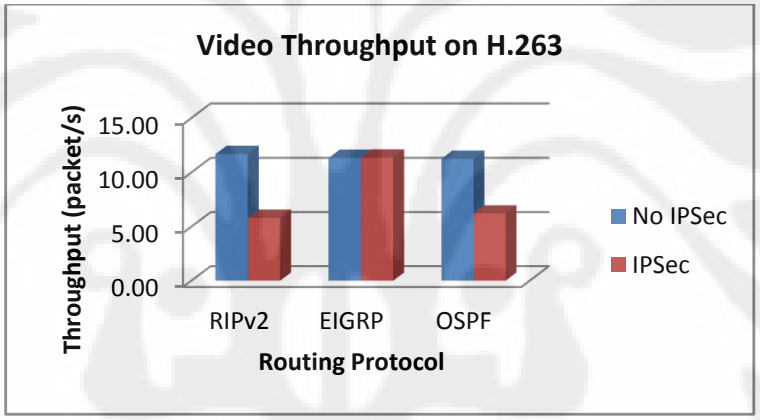
4.4.3.2 Kondisi Jaringan Tidak Stabil

Throughput audio pada *RIPv2*, *EIGRP* dan *OSPF* relatif sama sebelum implementasi *IPSec*, yaitu rata-rata 46 paket/detik, dimana *throughput EIGRP* relatif lebih kecil. Setelah implementasi *IPSec*, throughput *EIGRP* mengalami penurunan yang lebih besar daripada *RIPv2* dan *OSPF*, yaitu sekitar 13%, sedangkan *throughput RIPv2* turun 11% dan *OSPF* turun 9%. Dari analisa yang telah dilakukan terhadap audio dengan codec G.711 pada saat *video conference*, diperoleh karakteristik semakin kecil *delay*, semakin kecil *jitter*, semakin kecil *packet loss*, semakin besar *throughput* dan demikian sebaliknya.



Gambar 4.17 *Throughput* rata-rata G.711 pada kondisi jaringan tidak stabil

Pada dasarnya *throughput* berhubungan erat dengan *packet loss*, dimana semakin besar *packet loss* maka *throughput*nya akan semakin kecil. Hal ini yang terlihat pada trafik audio pada kondisi jaringan tidak stabil.



Gambar 4.18 *Throughput* rata-rata H.263 pada kondisi jaringan tidak stabil

Namun hal tersebut tidak berlaku pada trafik video ketika melakukan *video conference*, dimana berbeda dengan trafik audio, trafik video mempunyai rate paket yang ekstrim dan ukuran paket data yang sangat beragam, hal ini dapat dilihat pada paket *RTP* (H.263) untuk *codec* video yang tertangkap oleh wireshark. Pada grafik yang ditunjukkan oleh Gambar 4.18, *throughput* video pada *RIPv2*, *EIGRP* dan *OSPF* relatif sama sebelum implementasi *IPSec*, yaitu rata-rata 11 paket/detik, namun setelah implementasi *IPSec throughput RIPv2* dan *OSPF* menurun sebesar 50% dan 56%, sedangkan *throughput EIGRP* relatif sama seperti sebelum implementasi *IPSec*. Jika pada jaringan stabil diperoleh karakteristik semakin besar *delay*, *jitter* semakin kecil, dan *throughput* semakin kecil, maka hal tersebut tidak berlaku pada saat kondisi jaringan tidak stabil.

Secara keseluruhan, dapat diketahui bahwa pengaruh adanya implementasi *IPSec* dan perubahan kondisi jaringan lebih besar pada trafik video daripada trafik suara, kualitas suara menurun, namun tidak sebesar kualitas video yang sangat menurun pada saat diimplementasikan *IPSec* dalam kondisi jaringan yang tidak stabil. Hal ini disebabkan oleh trafik video memiliki rate paket yang lebih ekstrim dengan nilai rata-rata *Maximum Transmission Unit (MTU)* yang lebih tinggi daripada trafik suara. Di dalam jaringan, ukuran-ukuran paket pada trafik *video conference* sangat beragam.

Dilihat dari nilai *delay* yang terukur baik sebelum maupun setelah implementasi *IPSec*, pada kondisi jaringan stabil dan tidak stabil, *EIGRP* menempati urutan pertama yang memiliki kinerja paling bagus berdasarkan analisa yang telah diuraikan diatas, menempati urutan kedua adalah *OSPF* dan ketiga adalah *RIPv2*. *Delay* adalah parameter yang paling berperan dalam menentukan kinerja *routing* protokol dengan aplikasi *video conference* yang memang sensitif terhadap *delay*. *Delay* dan *jitter* saling melengkapi, *jitter* dapat menentukan besarnya *throughput* untuk trafik suara dan video pada kondisi jaringan stabil, sedangkan pada kondisi jaringan tidak stabil, *packet loss* yang menentukan besarnya *throughput* untuk trafik suara, tetapi tidak untuk trafik video yang memiliki ukuran paket yang beragam dalam jaringan.

4.5 Pengujian Kualitas Gambar dan Suara

Kualitas gambar dan suara merupakan faktor utama yang secara langsung bisa dilihat dan didengar oleh pengguna untuk menyatakan kepuasan atas apa yang dihasilkan dari *video conference* dan salah satu cara yang digunakan untuk penentuan kualitas gambar dan suara adalah dengan menggunakan penilaian subyektif dari responden, dengan memberikan penilaian kualitas baik gambar dan suara yang didefinisikan dengan skala 1 s/d 4 sebagai *Mean Opinion Score (MOS)* sebagai berikut :

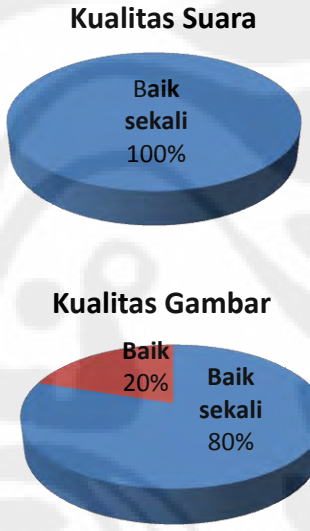
Nilai : 4 → Sangat baik
3 → Baik
2 → Cukup baik
1 → Kurang baik

Sesuai dengan analisa yang telah dilakukan, bahwa *EIGRP* mempunyai kinerja yang lebih bagus daripada *RIPv2*, dan *OSPF* baik sebelum maupun setelah implementasi *IPSec*, Untuk memperkuat analisa tersebut, kinerja *routing EIGRP* akan dievaluasi lagi dengan penilaian kualitas gambar dan suara dari *video conference* yang dilakukan oleh responden secara subyektif. Nilai *MOS* diperoleh dari 10 orang responden dan hasilnay dapat dilihat pada Tabel 4.5 dan Tabel 4.6.

Pada saat melakukan *video conference* dengan 2 *user,* dari Tabel 4.5 dan grafik pada Gambar 4.19 dapat diketahui bahwa 100% responden sepakat bahwa kualitas *video conference* untuk suara adalah sangat baik, sedangkan untuk gambar 80% responden berpendapat bahwa kualitasnya sangat baik, sedangkan 20% lainnya menilai kualitas gambar baik saja.

Tabel 4.5 Nilai *MOS* dari responden untuk *video conference* 2 *user*

No	Responden	Suara	Gambar
1	User 1	4	4
2	User 2	4	4
3	User 3	4	4
4	User 4	4	4
5	User 5	4	4
6	User 6	4	4
7	User 7	4	4
8	User 8	4	3
9	User 9	4	3
10	User 10	4	4

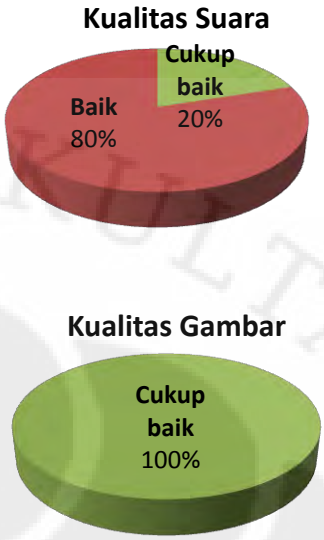


Gambar 4.19 Prosentase nilai *MOS* untuk kualitas suara dan gambar *video conference* 2 *user*

Pada saat *video conference* dengan 3 user, terlihat pada Tabel 4.6 dan grafik yang ditunjukkan oleh Gambar 4.20, bahwa 80% responden berpendapat bahwa kualitas suara yang dihasilkan baik saja, dan 20% lainnya menilai cukup baik. Sedangkan untuk kualitas gambar, 100% responden sepakat bahwa kualitas gambar cukup baik.

Tabel 4.6 Nilai *MOS* dari responden untuk *video conference* 3 *user*

No	Responden	Suara	Gambar
1	User 1	2	2
2	User 2	3	2
3	User 3	3	2
4	User 4	2	2
5	User 5	3	2
6	User 6	3	2
7	User 7	3	2
8	User 8	3	2
9	User 9	3	2
10	User 10	3	2



Gambar 4.20 Prosentase nilai *MOS* untuk kualitas suara dan gambar *video conference* 3 user

Data *MOS* diatas cukup untuk dijadikan acuan terhadap penilaian kualitas gambar dan suara dari *video conference* yang dilakukan, sekaligus sebagai data pendukung atas analisa yang telah dilakukan terhadap parameter-parameter jaringan yang terukur, sehingga diperoleh kesimpulan bahwa *EIGRP* adalah *routing* protokol yang paling optimal untuk diimplementasikan pada jaringan *MPLS* dengan tunnel *IPSec* pada skala *network* yang kecil dan *bandwidth* yang terbatas.

BAB 5

KESIMPULAN

Dari hasil analisa dan pengujian yang telah dilakukan, dapat disimpulkan sebagai berikut :

- Jaringan *MPLS* saja tidak cukup aman untuk melakukan komunikasi data, karena trafik *video conference* yang melintasi jaringan masih dapat dilihat sebagai paket *RTP*.
- Implementasi *IPSec* membuat *video conference* yang dilakukan menjadi aman, karena semua trafik data *video conference* terlihat sebagai paket data *ESP*.
- *Delay* video pada *EIGRP* mengalami kenaikan paling kecil setelah implementasi *IPSec,* yaitu sebesar 8%. Sedangkan *delay* audio *EIGRP* relatif sama dengan *RIPv2* dan *OSPF* baik sebelum maupun setelah implementasi *IPSec*, yaitu sekitar 20 ms.
- Penambahan *user* menyebabkan penurunan kualitas *video conference*, *delay* video pada *EIGRP* rata-rata paling kecil daripada *RIPv2* dan *OSPF* setelah implementasi *IPSec*, yaitu 215 ms, masih dibawah standar *delay* maksimum *ITU-T* yang dapat diterima pengguna. Kenaikan d*elay* audio pada *EIGRP* relatif sama dengan *RIPv2* dan *OSPF* setelah implementasi *IPSec* , yaitu sekitar 14%.
- *Routing protocol* yang paling optimal dan paling efektif untuk diimplementasikan pada jaringan *VPN* berbasis *MPLS* dan *IPSec* adalah *EIGRP*.

DAFTAR REFERENSI

[1] James Reagan. (2002). CCIP™ MPLS Study Guide. Alameda CA : SYBEX Inc.

[2] Anonymous. Cisco IOS Enterprise VPN Configuration Guide. *Site-to-Site And Extranet VPN Business Scenarioss* (Chapter 3).

[3] Kevin Dooley, Ian Brown. (2006). Cisco IOS Cookbook (2nd Edition). O'Reilly.

[4] Scott Hogg. (2002). EIGRP and OSPF Comparison. Project Number 02.

[5] Ruta Jankuiniene, Ieva jankuinaite. (2009). Route Creation Influence DMVPN QoS. IEEE Computer Society Journal.

[6] Deris Stiawan. Membangun Interkoneksi VPN dengan solusi Hardware-Based dan Jaringan IIX.

[7] Kuncoro Wastuwibowo. (2003). Jaringan MPLS Whitepaper. Versi 1.2. Telkom.info.

[8] Cisco Systems Learning. (2006). Implementing Secure Converged Wide Area Networks (Volume 1). San Jose : Cisco Systems Inc.

[9] Rudi Sutiono. CH4. - RIPv2 (CCNP 1 version 3.0). Cabrillo College.

[10] Rosen, Eric. et al. "Multiprotocol Label Switching Architecture". RFC-3031 IETF. January 2001. URL:http://www.ietf.org/rfc/rfc3031.txt(09/18/2004).

[12] Cisco Systems, Routing and Routing Protocols (Module 6). CCNA-Semester 2. NETPRO-ITI Academy.

[13] Scribd. Konsep Video Conference. Diakses pada tanggal 20 Mei 2010. URL:http://www.scribd.com/doc/7310023/BAB-II

[14] Cisco Systems Learning. (2004). Interconnecting Cisco Network Devices (version 2.2). San Jose : Cisco System Inc.

[15] ITU-T H-series Ss. "Realtime Multimedia Conferencing".

LAMPIRAN

Lampiran 1 : Konfigurasi *VPN MPLS* dengan *IPSec* Mode *Tunnel* Menggunakan *Routing Protocol RIPv2, EIGRP* dan *OSPF*

1. Konfigurasi *VPN MPLS* dengan *IPSec - RIPv2*

1.1 Router *Head Office*

```
Head-Office#sh run
Building configuration...
Current configuration : 1427 bytes
!
version 12.4
no service pad
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname Head-Office
!
boot-start-marker
boot-end-marker
!
enable password password
!
no aaa new-model
!
resource policy
!
ip subnet-zero
ip cef
!
crypto isakmp policy 1
encr 3des
authentication pre-share
group 2
crypto isakmp key p4ssw0rd address 10.10.13.2
!
crypto ipsec transform-set finalproject esp-3des esp-sha-hmac
!
crypto map skripsi 1 ipsec-isakmp
set transform-set finalproject
match address JKT-to-SBY
!
interface Loopback0
 ip address 10.10.12.1 255.255.255.255
!
interface FastEthernet1
switchport access vlan 2
!
interface FastEthernet2
!
interface FastEthernet3
!
interface FastEthernet4
description "Connection to PE-JKT"
ip address 10.10.11.2 255.255.255.252
duplex auto
speed auto
crypto map skripsi
!
interface Vlan1
no ip address
!
interface Vlan2
ip address 10.10.10.1 255.255.255.0
!
router rip
version 2
network 10.0.0.0
!
ip classless
!
no ip http server
no ip http secure-server
!
ip access-list extended JKT-to-SBY
permit ip 10.10.10.0 0.0.0.255 10.10.14.0 0.0.0.255
!
snmp-server community skripsi RW
!
control-plane
!
line con 0
no modem enable
line aux 0
line vty 0 4
password bismillah
```

```
login
!
scheduler max-task-time 5000
end
```

1.2 Router PE-JKT

```
PE-JKT#sh run
Building configuration...
Current configuration : 1709 bytes
!
version 12.3
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname PE-JKT
!
boot-start-marker
boot-end-marker
!
enable password password
!
no aaa new-model
ip subnet-zero
!
ip vrf company-a
rd 65000:1
route-target export 65000:1
route-target import 65000:1
!
ip cef
!
interface Loopback0
ip address 200.20.20.100 255.255.255.255
!
interface FastEthernet0/0
description "Connection to CORE"
ip address 200.20.20.1 255.255.255.248
duplex auto
speed auto
tag-switching ip
!
interface FastEthernet0/1
description "Connection to Head-Office"
ip vrf forwarding company-a
ip address 10.10.11.1 255.255.255.252
rate-limit input 256000 256000 256000 conform-action continue exceed-action drop
rate-limit output 256000 256000 256000 conform-action continue exceed-action drop
duplex auto
speed auto
!
router ospf 100
log-adjacency-changes
network 200.20.20.0 0.0.0.7 area 0
network 200.20.20.100 0.0.0.0 area 0
!
router ospf 101 vrf company-a
log-adjacency-changes
redistribute bgp 65000 subnets
network 10.10.11.0 0.0.0.3 area 0
!
router bgp 65000
no synchronization
bgp log-neighbor-changes
neighbor 200.20.20.102 remote-as 65000
neighbor 200.20.20.102 update-source Loopback0
neighbor 200.20.20.102 next-hop-self
no auto-summary
!
address-family vpnv4
neighbor 200.20.20.102 activate
neighbor 200.20.20.102 send-community both
exit-address-family
!
address-family ipv4 vrf company-a
redistribute ospf 101
no auto-summary
no synchronization
exit-address-family
!
ip http server
ip classless
!
line con 0
line aux 0
line vty 0 4
password bismillah
login
!
end
```

1.3 Router *Core*

```
CORE#sh run
Building configuration...
Current configuration : 904 bytes
!
version 12.3
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
```

```
!
hostname CORE
!
boot-start-marker
boot-end-marker
!
enable password password
!
no aaa new-model
ip subnet-zero
!
ip cef
!
interface Loopback0
 ip address 200.20.20.101 255.255.255.255
!
interface FastEthernet0/0
description "Connection to PE-JKT"
ip address 200.20.20.2 255.255.255.248
duplex auto
speed auto
tag-switching ip
!
interface FastEthernet0/1
description "Connection to PE-SBY"
ip address 200.20.20.9 255.255.255.252
duplex auto
speed auto
tag-switching ip
!
log-adjacency-changes
router ospf 100
network 200.20.20.0 0.0.0.7 area 0
network 200.20.20.8 0.0.0.3 area 0
network 200.20.20.101 0.0.0.0 area 0
!
ip http server
ip classless
!
line con 0
line aux 0
line vty 0 4
password bismillah
login
!
End
```

1.4 Router PE-SBY

```
PE-JKT#sh run
Building configuration...
Current configuration : 1495 bytes
!
interface FastEthernet0
 switchport access vlan 2
!
version 12.3
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname PE-JKT
!
boot-start-marker
boot-end-marker
!
enable password password
!
no aaa new-model
ip subnet-zero
!
ip vrf company-a
rd 65000:1
route-target export 65000:1
route-target import 65000:1
!
ip cef
!
interface Loopback0
ip address 200.20.20.100 255.255.255.255
!
interface FastEthernet0/0
description "Connection to CORE"
ip address 200.20.20.1 255.255.255.252
duplex auto
speed auto
tag-switching ip
!
interface FastEthernet0/1
description "Connection to Head-Office"
ip vrf forwarding company-a
ip address 10.10.11.1 255.255.255.252
duplex auto
speed auto
!
router rip
version 2
network 200.20.20.0
!
address-family ipv4 vrf company-a
redistribute bgp 65000 metric transparent
network 10.0.0.0
no auto-summary
version 2
exit-address-family
!
router bgp 65000
no synchronization
bgp log-neighbor-changes
neighbor 200.20.20.102 remote-as 65000
neighbor 200.20.20.102 update-source
Loopback0
```

```
 neighbor 200.20.20.102 next-hop-self
 no auto-summary
 !
 address-family vpnv4
 neighbor 200.20.20.102 activate
 neighbor 200.20.20.102 send-community
 both
 exit-address-family
 !
 address-family ipv4 vrf company-a
 redistribute rip
 no auto-summary
 no synchronization
 exit-address-family
 !
 ip http server
 ip classless
 !
 line con 0
 line aux 0
 line vty 0 4
 password bismillah
 login
 !
 End
```

1.3 Router *Branch Office*

```
Branch-Office#sh run
Building configuration...
Current configuration : 1454 bytes
!
version 12.4
no service pad
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname Branch-Office
!
boot-start-marker
boot-end-marker
!
enable password password
!
no aaa new-model
!
resource policy
!
ip subnet-zero
ip cef
!
crypto isakmp policy 1
 encr 3des
 authentication pre-share
 group 2
crypto isakmp key p4ssw0rd address
10.10.11.2
!
crypto ipsec transform-set finalproject esp-
3des esp-sha-hmac
!
crypto map skripsi 1 ipsec-isakmp
 set peer 10.10.11.2
 set transform-set finalproject
 match address SBY-to-JKT
!
interface Loopback0
 ip address 10.10.15.1 255.255.255.255
!
interface FastEthernet0
 switchport access vlan 2
!
interface FastEthernet1
 switchport access vlan 2
!
interface FastEthernet2
 switchport access vlan 2
!
interface FastEthernet3
!
interface FastEthernet4
description "Connetion to PE-SBY"
ip address 10.10.13.2 255.255.255.252
duplex auto
speed auto
crypto map skripsi
!
interface Vlan1
no ip address
!
interface Vlan2
ip address 10.10.14.1 255.255.255.0
!
router rip
version 2
network 10.0.0.0
!
ip classless
!
no ip http server
no ip http secure-server
!
ip access-list extended SBY-to-JKT
permit ip 10.10.14.0 0.0.0.255 10.10.10.0
0.0.0.255
!
snmp-server community skripsi RO
!
control-plane
!
line con 0
```

```
no modem enable
line aux 0
line vty 0 4
password bismillah
login
!
scheduler max-task-time 5000
end
```

2. Konfigurasi *VPN MPLS* dengan *IPSec – EIGRP*

2.1 Router *Head Office*

```
Head-Office#sh run
Building configuration...
Current configuration : 1436 bytes
!
version 12.4
no service pad
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname Head-Office
!
boot-start-marker
boot-end-marker
!
enable password password
!
no aaa new-model
!
resource policy
!
ip subnet-zero
ip cef
!
crypto isakmp policy 1
encr 3des
authentication pre-share
group 2
crypto isakmp key p4ssw0rd address 10.10.13.2
!
crypto ipsec transform-set finalproject esp-3des esp-sha-hmac
!
crypto map skripsi 1 ipsec-isakmp
set peer 10.10.13.2
set transform-set finalproject
match address JKT-to-SBY
!
interface Loopback0
ip address 10.10.12.1 255.255.255.255
!
interface FastEthernet0
switchport access vlan 2
!
interface FastEthernet1
switchport access vlan 2
!
interface FastEthernet2
!
interface FastEthernet3
!
interface FastEthernet4
description "Connection to PE-JKT"
ip address 10.10.11.2 255.255.255.252
duplex auto
speed auto
!
interface Vlan1
no ip address
!
interface Vlan2
ip address 10.10.10.1 255.255.255.0
!
router eigrp 100
network 10.0.0.0
auto-summary
!
ip classless
!
no ip http server
no ip http secure-server
!
ip access-list extended JKT-to-SBY
permit ip 10.10.10.0 0.0.0.255 10.10.14.0 0.0.0.255
!
snmp-server community skripsi RW
!
control-plane
!
line con 0
no modem enable
line aux 0
line vty 0 4
password bismillah
login
!
scheduler max-task-time 5000
end
```

2.1 Router PE-JKT

```
PE-JKT#sh run
Building configuration...
Current configuration : 1724 bytes
!
debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname PE-JKT
!
boot-start-marker
boot-end-marker
!
enable password password
!
no aaa new-model
ip subnet-zero
!
ip vrf company-a
rd 65000:1
route-target export 65000:1
route-target import 65000:1
!
ip cef
!
interface Loopback0
ip address 200.20.20.100 255.255.255.255
!
interface FastEthernet0/0
description "Connection to CORE"
ip address 200.20.20.1 255.255.255.248
duplex auto
speed auto
tag-switching ip
!
interface FastEthernet0/1
description "Connection to Head-Office"
ip vrf forwarding company-a
ip address 10.10.11.1 255.255.255.252
rate-limit input 256000 256000 256000 conform-action continue exceed-action drop
rate-limit output 256000 256000 256000
duplex auto
speed auto
!
router eigrp 100
network 200.20.20.0
auto-summary
!
address-family ipv4 vrf company-a
redistribute bgp 65000 metric 10000 100 255 1 1500
network 10.0.0.0
auto-summary
autonomous-system 100
eigrp router-id 200.20.20.100
exit-address-family
!
router bgp 65000
no synchronization
bgp log-neighbor-changes
neighbor 200.20.20.102 remote-as 65000
neighbor 200.20.20.102 update-source Loopback0
neighbor 200.20.20.102 next-hop-self
no auto-summary
!
address-family vpnv4
neighbor 200.20.20.102 activate
neighbor 200.20.20.102 send-community both
exit-address-family
!
address-family ipv4 vrf company-a
redistribute eigrp 100
no auto-summary
no synchronization
exit-address-family
!
ip http server
ip classless
!
line con 0
line aux 0
line vty 0 4
password bismillah
login
!
End
```

2.2 Router *Core*

```
CORE#sh run
Building configuration...
Current configuration : 807 bytes
!
version 12.3
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname CORE
!
boot-end-marker
!
enable password password
!
no aaa new-model
ip subnet-zero
```

```
!
ip cef
!
interface Loopback0
 ip address 200.20.20.101 255.255.255.255
!
interface FastEthernet0/0
 description "Connection to PE-JKT"
 ip address 200.20.20.2 255.255.255.248
 duplex auto
 speed auto
 tag-switching ip
!
interface FastEthernet0/1
 description "Connection to PE-SBY"
 ip address 200.20.20.9 255.255.255.252
 duplex auto
 speed auto
 tag-switching ip
!
router eigrp 100
 network 200.20.20.0
 auto-summary
!
ip http server
ip classless
!
line con 0
line aux 0
line vty 0 4
password bismillah
login
!
End
```

2.3 Router PE-SBY

```
PE-SBY#sh run
Building configuration...
Current configuration : 1727 bytes
!
version 12.3
boot-start-marker
boot-end-marker
!
enable password password
!
no aaa new-model
ip subnet-zero
!
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname PE-SBY
!
ip vrf company-a
 rd 65000:1
 route-target export 65000:1
 route-target import 65000:1
!
ip cef
!
interface Loopback0
 ip address 200.20.20.102 255.255.255.255
!
interface FastEthernet0/0
description "Connection to Branch-Office"
ip vrf forwarding company-a
ip address 10.10.13.1 255.255.255.252
rate-limit input 256000 256000 256000
conform-action continue exceed-action drop
rate-limit output 256000 256000 256000
conform-action continue exceed-action drop
duplex auto
speed auto
!
interface FastEthernet0/1
description "Connection to CORE"
ip address 200.20.20.10 255.255.255.252
duplex auto
speed auto
tag-switching ip
!
router eigrp 100
network 200.20.20.0
auto-summary
!
address-family ipv4 vrf company-a
redistribute bgp 65000 metric 10000 100
255 1 1500
network 10.0.0.0
auto-summary
autonomous-system 100
eigrp router-id 200.20.20.102
exit-address-family
!
router bgp 65000
no synchronization
bgp log-neighbor-changes
neighbor 200.20.20.100 remote-as 65000
neighbor 200.20.20.100 update-source
Loopback0
neighbor 200.20.20.100 next-hop-self
no auto-summary
!
```

```
address-family vpnv4
neighbor 200.20.20.100 activate
neighbor 200.20.20.100 send-community both
exit-address-family
!
address-family ipv4 vrf company-a
redistribute eigrp 100
no auto-summary
no synchronization
exit-address-family
!
ip http server
ip classless
!
line con 0
line aux 0
line vty 0 4
password bismillah
login
!
End
```

2.4 Router *Branch Office*

```
Branch-Office#sh run
Building configuration...
Current configuration : 1552 bytes
!
version 12.4
no service pad
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname Branch-Office
!
boot-start-marker
boot-end-marker
!
enable password password
!
no aaa new-model
!
monitor session 1 source interface Fa0 - 1
monitor session 1 destination interface Fa2
!
resource policy
!
ip subnet-zero
ip cef
!
crypto isakmp policy 1
 encr 3des
 authentication pre-share
 group 2
crypto isakmp key p4ssw0rd address 10.10.11.2
!
crypto ipsec transform-set finalproject esp-3des esp-sha-hmac
!
crypto map skripsi 1 ipsec-isakmp
set peer 10.10.11.2
set transform-set finalproject
match address SBY-to-JKT
!
interface Loopback0
ip address 10.10.15.1 255.255.255.255
!
interface FastEthernet0
switchport access vlan 2
!
interface FastEthernet1
switchport access vlan 2
!
interface FastEthernet2
switchport access vlan 2
!
interface FastEthernet3
!
interface FastEthernet4
description "Connetion to PE-SBY"
ip address 10.10.13.2 255.255.255.252
duplex auto
speed auto
crypto map skripsi
!
interface Vlan1
no ip address
!
interface Vlan2
ip address 10.10.14.1 255.255.255.0
!
router eigrp 100
network 10.0.0.0
auto-summary
!
ip classless
!
no ip http server
no ip http secure-server
!
ip access-list extended SBY-to-JKT
permit ip 10.10.14.0 0.0.0.255 10.10.10.0 0.0.0.255
!
snmp-server community skripsi RO
```

```
!
control-plane
!
line con 0
no modem enable
line aux 0
line vty 0 4
password bismillah
 login
!
scheduler max-task-time 5000
end
```

3. Konfigurasi *VPN MPLS* dengan *IPSec – OSPF*

3.1 Router *Head Office*

```
Head-Office#sh run
Building configuration...
Current configuration : 1533 bytes
!
version 12.4
no service pad
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname Head-Office
!
boot-start-marker
boot-end-marker
!
enable password password
!
no aaa new-model
!
resource policy
!
ip subnet-zero
ip cef
!
crypto isakmp policy 1
encr 3des
authentication pre-share
group 2
crypto isakmp key p4ssw0rd address 10.10.13.2
!
crypto ipsec transform-set finalproject esp-3des esp-sha-hmac
!
crypto map skripsi 1 ipsec-isakmp
set peer 10.10.13.2
set transform-set finalproject
match address JKT-to-SBY
!
interface Loopback0
ip address 10.10.12.1 255.255.255.255
!
interface FastEthernet0
switchport access vlan 2
!
interface FastEthernet1
switchport access vlan 2
!
interface FastEthernet2
!
interface FastEthernet3
!
interface FastEthernet4
description "Connection to PE-JKT"
ip address 10.10.11.2 255.255.255.252
duplex auto
speed auto
crypto map skripsi
!
interface Vlan1
no ip address
!
interface Vlan2
ip address 10.10.10.1 255.255.255.0
!
router ospf 101
log-adjacency-changes
network 10.10.10.0 0.0.0.255 area 0
network 10.10.11.0 0.0.0.3 area 0
network 10.10.12.1 0.0.0.0 area 0
!
ip classless
!
no ip http server
no ip http secure-server
!
ip access-list extended JKT-to-SBY
permit ip 10.10.10.0 0.0.0.255 10.10.14.0 0.0.0.255
!
snmp-server community skripsi RW
!
control-plane
!
line con 0
no modem enable
```

```
line aux 0
line vty 0 4
password bismillah
login
!
scheduler max-task-time 5000
end
```

3.2 Router PE-JKT

```
PE-JKT#sh run
Building configuration...
Current configuration : 1709 bytes
!
version 12.3
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname PE-JKT
!
boot-start-marker
boot-end-marker
!
enable password password
!
no aaa new-model
ip subnet-zero
!
ip vrf company-a
rd 65000:1
route-target export 65000:1
route-target import 65000:1
!
ip cef
!
interface Loopback0
ip address 200.20.20.100 255.255.255.255
!
interface FastEthernet0/0
description "Connection to CORE"
ip address 200.20.20.1 255.255.255.248
duplex auto
speed auto
tag-switching ip
!
interface FastEthernet0/1
description "Connection to Head-Office"
ip vrf forwarding company-a
ip address 10.10.11.1 255.255.255.252
rate-limit input 256000 256000 256000
conform-action continue exceed-action
conform-action continue exceed-action
drop
duplex auto
speed auto
!
router ospf 100
log-adjacency-changes
network 200.20.20.0 0.0.0.7 area 0
network 200.20.20.100 0.0.0.0 area 0
!
router ospf 101 vrf company-a
log-adjacency-changes
redistribute bgp 65000 subnets
network 10.10.11.0 0.0.0.3 area 0
!
router bgp 65000
no synchronization
bgp log-neighbor-changes
neighbor 200.20.20.102 remote-as
65000
neighbor 200.20.20.102 update-source
Loopback0
neighbor 200.20.20.102 next-hop-self
no auto-summary
!
address-family vpnv4
neighbor 200.20.20.102 activate
neighbor 200.20.20.102 send-
community both
exit-address-family
!
address-family ipv4 vrf company-a
redistribute ospf 101
no auto-summary
no synchronization
exit-address-family
!
ip http server
ip classless
!
line con 0
line aux 0
line vty 0 4
password bismillah
login
!
End
```

3.3 Router Core

```
CORE#sh run
Building configuration...
Current configuration : 904 bytes
!
```

```
version 12.3
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname CORE
!
boot-start-marker
boot-end-marker
!
enable password password
!
no aaa new-model
ip subnet-zero
!
ip cef
!
interface Loopback0
 ip address 200.20.20.101 255.255.255.255
!
interface FastEthernet0/0
description "Connection to PE-JKT"
ip address 200.20.20.2 255.255.255.248
duplex auto
speed auto
tag-switching ip
!
interface FastEthernet0/1
description "Connection to PE-SBY"
ip address 200.20.20.9 255.255.255.252
duplex auto
speed auto
tag-switching ip
!
router ospf 100
log-adjacency-changes
network 200.20.20.0 0.0.0.7 area 0
network 200.20.20.8 0.0.0.3 area 0
network 200.20.20.101 0.0.0.0 area 0
!
ip http server
ip classless
!
line con 0
line aux 0
line vty 0 4
password bismillah
login
!
End
```

3.4 Router PE-SBY

```
PE-SBY#sh run
Building configuration...
Current configuration : 1712 bytes
!
version 12.3
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname PE-SBY
!
boot-start-marker
boot-end-marker
!
enable password password
!
no aaa new-model
ip subnet-zero
!
ip vrf company-a
rd 65000:1
route-target export 65000:1
route-target import 65000:1
!
ip cef
!
interface Loopback0
ip address 200.20.20.102 255.255.255.255
!
interface FastEthernet0/0
description "Connection to Branch-Office"
ip vrf forwarding company-a
ip address 10.10.13.1 255.255.255.252
rate-limit input 256000 256000 256000 conform-action continue exceed-action drop
rate-limit output 256000 256000 256000 conform-action continue exceed-action drop
duplex auto
speed auto
!
interface FastEthernet0/1
description "Connection to CORE"
ip address 200.20.20.10 255.255.255.252
duplex auto
speed auto
tag-switching ip
```

```
!
router ospf 100
log-adjacency-changes
network 200.20.20.8 0.0.0.3 area 0
network 200.20.20.102 0.0.0.0 area 0
!
router ospf 101 vrf company-a
log-adjacency-changes
redistribute bgp 65000 subnets
network 10.10.13.0 0.0.0.3 area 0
!
router bgp 65000
no synchronization
bgp log-neighbor-changes
neighbor 200.20.20.100 remote-as 65000
neighbor 200.20.20.100 update-source Loopback0
neighbor 200.20.20.100 next-hop-self
no auto-summary
!
address-family vpnv4
neighbor 200.20.20.100 activate
neighbor 200.20.20.100 send-community both
exit-address-family
!
address-family ipv4 vrf company-a
redistribute ospf 101
no auto-summary
no synchronization
exit-address-family
!
ip http server
ip classless
!
line con 0
line aux 0
line vty 0 4
password bismillah
login
!
End
```

3.5 Router *Branch Office*

```
Branch-Office#sh run
Building configuration...

Current configuration : 1649 bytes
!
version 12.4
no service pad
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname Branch-Office
!
boot-start-marker
boot-end-marker
!
enable password password
!
no aaa new-model
!
monitor session 1 source interface Fa0 - 1
monitor session 1 destination interface Fa2
!
resource policy
!
ip subnet-zero
ip cef
!
crypto isakmp policy 1
encr 3des
authentication pre-share
group 2
crypto isakmp key p4ssw0rd address 10.10.11.2
!
crypto ipsec transform-set finalproject esp-3des esp-sha-hmac
!
crypto map skripsi 1 ipsec-isakmp
set peer 10.10.11.2
set transform-set finalproject
match address SBY-to-JKT
!
interface Loopback0
ip address 10.10.15.1 255.255.255.255
!
interface FastEthernet0
switchport access vlan 2
!
interface FastEthernet1
switchport access vlan 2
!
interface FastEthernet2
switchport access vlan 2
!
interface FastEthernet3
!
interface FastEthernet4
description "Connetion to PE-SBY"
ip address 10.10.13.2 255.255.255.252
duplex auto
speed auto
```

```
crypto map skripsi
!
interface Vlan1
 no ip address
!
interface Vlan2
 ip address 10.10.14.1 255.255.255.0
!
router ospf 101
log-adjacency-changes
network 10.10.13.0 0.0.0.3 area 0
network 10.10.14.0 0.0.0.255 area 0
network 10.10.15.0 0.0.0.0 area 0
!
ip classless
!
no ip http server
no ip http secure-server
!
ip access-list extended SBY-to-JKT
permit ip 10.10.14.0 0.0.0.255 10.10.10.0 0.0.0.255
!
snmp-server community skripsi RO
!
control-plane
!
line con 0
 no modem enable
line aux 0
line vty 0 4
 password bismillah
 login
!
scheduler max-task-time 5000
end
```