# BAB II
# TINJAUAN PUSTAKA DAN LANDASAN TEORI

## 2.1 Tinjauan pustaka

Teknologi *VPN* telah melalui beberapa evolusi sejak dimulai pada tahun 1996 oleh Gurdeep Sing Pall yang menemukan PPTP (*Proint to Point Tunneling Protocol*) sebagai metode implementasi *private network* secara *virtual*. Dengan perkembangan teknologi ini, memberikan pengguna mengkoneksikan kantor pusat ke cabang/*suppliers* dengan menggunakan infrastruktur telekomunikasi umum, terenkripsi dan terenkapsulasi. Selama beberapa tahun, berbagai tipe dari teknologi *VPN* telah muncul seperti *VPN for bussiness or personal* dengan menggunakan beberapa protokol berbeda, serta penggunanaan tipe enkripsi (*hashing*, *symmetric*, *asymmetric*) untuk mengamankan transaksi data (Alam, Biddut, Shafin, & Shariar, 2016).

Pembahasan mengenai *VPN* tidak lepas dari tingkat keamanan itu sendiri, menurut Jyothi & Reddy (2018) pada makalah penelitiannya menyebutkan bahwa *VPN* dapat melindungi jaringan lokal dengan menunjukkan variasi *encryption*, *authentication*, dan *integrity algorithms* yang sampai saat ini belum ada standar yang pasti oleh lembaga terkait tingkat keamanan karena *VPN* secara penuh belum dieksploitasi. Dalam perancangan dan penerapan *vpn network* dan *failover VPN* juga pernah dilakukan dalam beberapa penelitian sebelumnya sebagai berikut:

- Wiwin Sulistyo, S.T., M.Kom dan Yudhi Trihandian (2016) dalam artikel ilmiahnya berjudul simulasi *failover* pada protokol *routing*. Artikel ini membahas bagaimana mengimplementasikan *Failover link* pada perancangan *backup main link* yang mengalami gangguan dan merancangan jaringan *VPN* pada jalur cadangan menggunakan simulasi *packet tracer*.
- Agni Isador Harsapranata (2014) pada jurnal berjudul implementasi *failover* menggunakan jaringan *VPN* dan Metronet pada Astridogroup indonesia, pada penelitiannya mengimplementasikan teknologi metronet *Fiber Optik*

dan *VPN* sebagai jalur cadangan sebuah *gateway* yang apabila koneksi salah satu terdapat kegagalan maka *backup link* yang mengatur untuk *failover* tersebut.

- Ceisar Maulana Shabirin (2014) dalam skripsinya analisis implementasi *routing protocol authentication* pada jaringan *MPLS-L3VPN*. Analisis ini membahas masalah keamanan pada sisi integrity dari *main-in-the-middle-attack* saat menggunakan aplikasi *VoIP* yang menggunakan sistem autentikasi di setiap *peer MPLS*, dengan menggunakan teknologi *routing protocol authentication*. Hasil dari pengujian adalah informasi *routing* sangat mudah untuk didapatkan, sehingga membutuhkan autentikasi pada setiap *header*-nya yang kemudian dienkripsi dengan fungsi *hash MD5*.
- Faycal Bensalah, Najib El Kamoun dan Ayoub BAHNASSE (2017) dalam Jurnal penelitiannya berjudul "Evaluation of *tunnel* layer impact on VOIP performances (IP-MPLS-MPLS VPN-MPLS VPN IPsec)", menganalisis kinerja beberapa teknologi *MPLS VPN* yang di *protect* menggunakan *IPsec*. Pada penelitian ini melakukan studi pengujian skalabilitas dan dampak peroforma *tunnel layer* menggunakan *GNS3* dengan meningkatkan *load* dan variasi teknologi *MPLS VPN*. Keluaran dari pengujian berupa hasil pengukuran performa yang meggunakan trafik *VoIP* sebagai aplikasi pengujiannya.
- Abdullah Al Mamun, Tarek R. Sheltami, Hassan Ali, Sultan Anware (2016) dalam penelitiannya "Performance Evaluation of Routing Protocols for Video Conference over MPLS VPN Network", menganalisis kinerja perbandingan performa dari trafik video antara dua protokol *routing* menggunakan *GNS3* dan *OPNET Modelet 14.5* untuk melakukan berbagai simulasi skenario yang berbeda dan pengukuran nilai *metric* seperti *delay*, *jitter*, dan *Mean Opinion Score*. Hasil dari penelitian adalah menunjukkan bahwa *OSPF* dan *BGP-MPLS VPN* memberikan performa terbaik pada aplikasi video *conference*.
- K. Sandhya dan V. Kakulapati (2018) pada jurnalnya dengan judul "Establishing Secure Enterprise Network Routing protocols by using

DMPVN". Penelitian ini membahas bagaimana pembangunan *DMVPN* dan penggunaan dari dynamic *routing protocol* (*OSPF* dan *EIGRP*), juga melakukan simulasi menggunakan *GNS3* dengan tujuan mendapatkan pengukuran protokol *routing* yang tepat dalam penggunaan di perusahaan atau *enterprise network*. Hasil dari pengujian adalah *EIGRP* lebih kompatibel dibandingkan dengan *OSPF* karena tidak proporsi dan menjadi kompleks karena area yang dibatasi oleh *DMVPN* sehingga, lebih disarankan menggunakan *EIGRP* pada larger *network environment* dan juga *small network environment* ketika vendor jaringan bukan dari Cisco.

- Mojammad Rizal, Arini, Siti Ummi Masruroh (2018) pada penelitian evaluasi kinerja jaringan *DMVPN* menggunakan protokol *routing RIPv2*, *OSPF*, *EIGRP* dengan *BGP*, membahas protokol *routing* yang digunakan dengan algoritme yang berbeda-beda untuk mendapatkan nilai *QoS* dan *network covergence*. Hasil dari penelitian ini adalah nilai *QoS* terbaik ada *phase 1* adalah *EIGRP-BGP*, *phase 2* adalah *EIGRP-BGP* dan *phase* 3 adalah *RIPv2-BGP* sehingga kesimpulan secara keseluruhannya adalah *EIGRP-BGP* merupakan kombinasi *routing* protokol terbaik untuk *DMVPN*.

Dari hasil penelitian-penelitian yang telah dijabarkan tersebut, terdapat beberapa kesamaan mendasar dengan penelitian yang penulis lakukan, antara lain adalah pertama implementasi *failover link* pada perancangan mitigasi gangguan *main link* di perusahaan/kantor pusat ke kantor cabang. Kedua, penerapan teknologi *MPLS-L3VPN* sebagai *secure connection* pada *tunneling layer* beserta performa pengaplikasian video *conference* yang menjadi objek studi kasus pada penelitian ini. Dan ketiga adalah membangun teknologi *DMVPN* sebagai *secure connection* dengan menggunakan vendor jaringan Cisco pada objek studi kasus penelitian yang menjadi pokok pembahasan dalam penerapakan teknologi *failover link*.

Sementara itu, perbedaan penelitian yang penulis lakukan dengan penelitian-penelitian yang telah dipaparkan diatas adalah penelitian ini lebih membicarakan bagaimana perancangan *DMVPN* sebagai *failover VPN link* dari kantor pusat ke

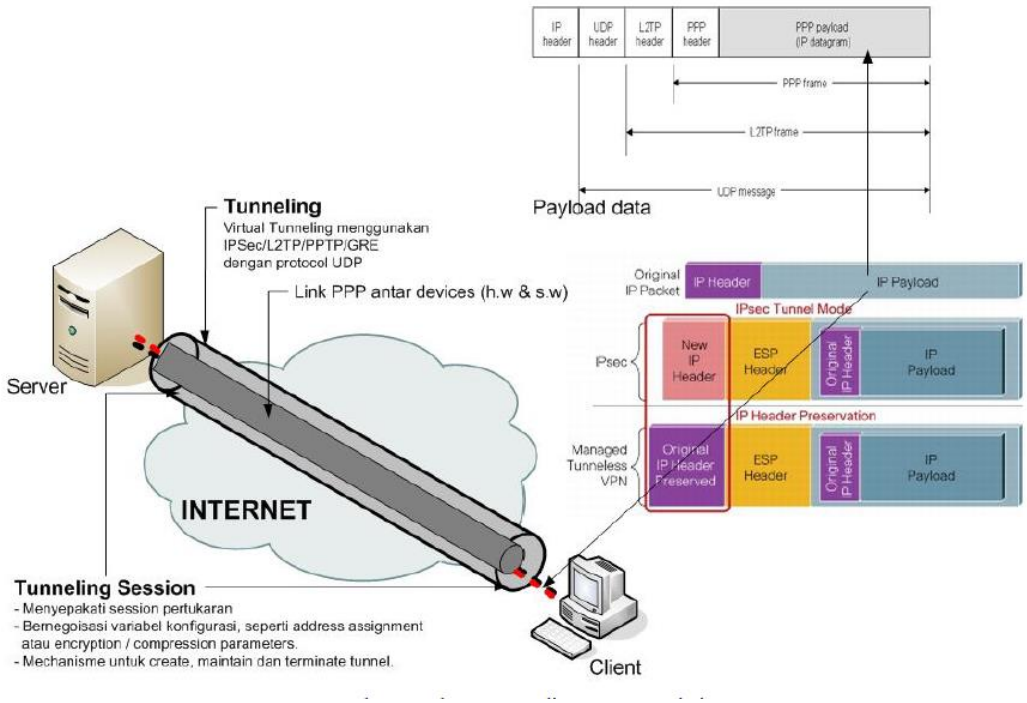
kantor cabang melalui WAN publik/*internet* sebagai alternatif dari *main link* yang menggunakan *MPLS-L3VPN* untuk mencapai *high avaibility* dalam kebutuhan komunikasi data yang aman. Selain itu, dibangun sebuah Video *streaming application* pada penelitian ini untuk mendapatkan pengukuran nilai *QoS* dan mengamati proses pergantian *VPN link* pada simulasi yang dibangun. Pada penggunaan protokol *routing* yang akan di terapkan pada teknologi *MPLS-L3VPN* dan *DMVPN* adalah *BGP* untuk *main link* dan *EIGRP* untuk jalur cadangan yang telah di analisis pengukuran pada performa masing-masing dari penelitian sebelumnya.

## 2.2 Landasan teori

### 2.2.1 *VPN*

*Virtual Private Network (VPN)* merupakan suatu koneksi antar dua jaringan yang dibuat untuk mengoneksikan kantor pusat, kantor cabang, *telecommuters*, *suppliers*, dan rekan bisnis lainnya, ke dalam suatu jaringan dengan menggunakan infrastruktur telekomunikasi umum dan menggunakan metode enkripsi tertentu sebagai media pengamanannya (Kevin, 2001).

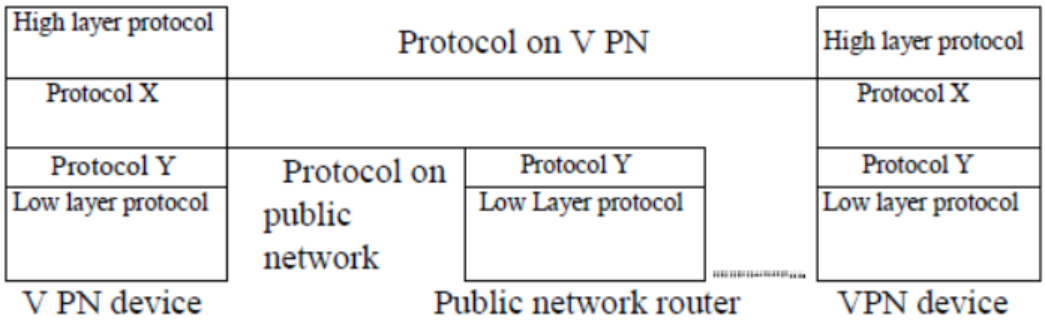
Jaringan *VPN* dikoneksikan oleh penyedia jasa komunikasi (*Service Porvider*) melalui *route*-nya ke *router-router* lain dengan menggunakan jalur *internet* yang telah di enkripsi di antara duat titik, seperti yang ditunjukkan pada gambar 2.1.




Gambar 2. 1 Skema *Tunneling* & *Encapsulation VPN* (sumber ui.ac.id)

Sistem keamanan di *VPN* menggunakan beberapa metode lapisan sistem keamanan, di antaranya:

a.) Metode *tunnel*ing




Gambar 2. 2 Arsitektur protokol *VPN* (sumber *Research on Tunneling Techniques in Virtual Private Networks*)

*Tunneling* sama dengan seperti *encapsulation*, dengan membuat *virtual tunnel* melalui WAN publik menggunakan protokol. Seperti contoh sebuah protokol X dienkapsulasikan kedalam protokol Y ketika proses pengiriman,

sehingga protokol X tersebut menjadi transparan ke jaringan publik. Arsitektur protokol dari *VPN* yang diterapkan menggunakan mekanisme *tunnel*ing yang diilustrasikan di gambar 2.2. dari gambar tersebut protokol X merupakan protokol yang dienkapsulasi sedangkan protokol Y merupakan protokol yang mengenkapsulasikannya. Secara umum ketika mengenkapsulasikan, pada protokol *tunnel*ing tertentu menggunakan *IP* sebagai protokol enkapsulasi yang dinamakan *IP tunneling protocol*. Saat ini sudah ada beberapa yang menawarkan *IP tunneling protocol* seperti *PPTP, L2TP, GRE atau IPsec* (Amankatiyar , Hemantjain , JayeshSurana, Soni, & Vishwakarma, 2017).

b.) Metode enkripsi

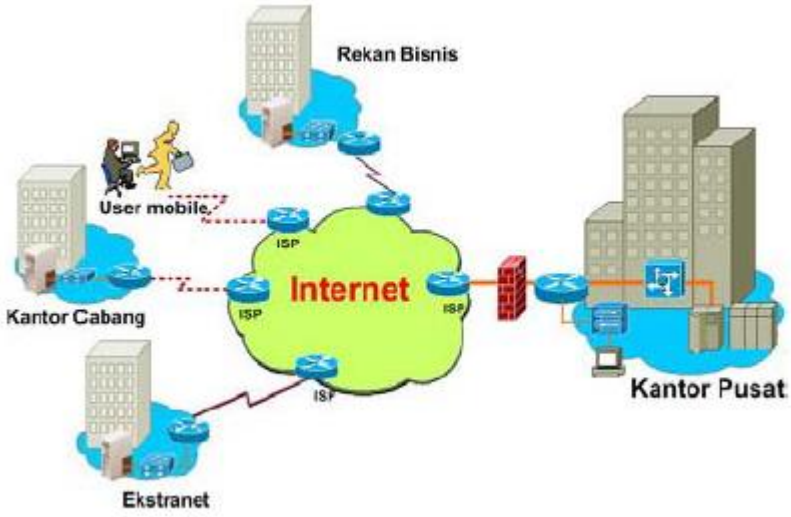
Merupakan sebuah *generate key* untuk mengenkripsikan data digital sehingga data tidak dapat di akses oleh seseorang yang tidak memiliki izin. Proses dari enkripsi *VPN* sendiri bergantung pada standar dan penggunaan *software VPN*. Data yang dienkripsikan melalui algoritme kriptografi tertentu seperti *DES, 3DES, AES, BLOWFISH,* dan *RC4* memerlukan *key* yang sama digunakan untuk mendekripsikan data tersebut. Semakin panjang *key* yang digunakan, maka diperlukan juga waktu yang mana untuk memecahkan *key* tersebut dan enkripsinya lebih kuat (Alam, Biddut, Shafin, & Shariar, 2016)

c.) Metode otentikasi

Otentikasi memastikan identitas pihak-pihak yang akan berkomunikasi melalui proses pemeriksaan sandi ataupun *key file* seperti *Remote Access Dial in User Service (RADIUS)* dan Digital Certificates (Alam, Biddut, Shafin, & Shariar, 2016).

d.) Integritas data

*VPN* juga memberikan pemeriksaan integritas data yang secara khususnya dilakukan menggunakan *message digest* untuk memastikan bahwa data tidak diubah selama data tersebut sedang dikirim (Alam, Biddut, Shafin, & Shariar, 2016).




Gambar 2. 3 *Access*, *Intranet* dan *Ekstranet VPN* (sumber ui.ac.id)

Ada tiga macam tipe interkoneksi VPN (Kevin, 2001) :

a.) *Access VPN*

*VPN* yang memberikan remote access ke jaringan intranet maupun extranet yang kinerja dan aksesnya yang sama seperti jaringan internal. Dengan menggunkan Access *VPN* memungkinkan pengguna dapat mengakses data dan berkomunikasi dengan perusahaan dari manapun dan kapan pun.

*b.) Intranet VPN*

*VPN* yang memberikan koneksi antara kantor pusat, kantor cabang dan *remote user* ke dalam jaringan internal melalui infrastruktur *dedicated* atau permanen.

*c.) Ekstranet VPN*

*VPN* yang menghubungkan jaringan internal perusahaan dengan jaringan internal perusahaan lain, pelanggan, *supplier* atau komunitas yang memerlukan komunikasi data melalui infrastrktur yang *dedicated* atau *permanen*.

### 2.2.2 *MPLS*

*MPLS* atau *Multiprotocol Label Switching* adalah perkembangan sistem komunikasi dari *circuit-switched* dan *packet-switched* yang memberikan teknologi pengiriman data pada jaringan utama atau *backbone* dengan kecepatan tinggi. *IETF* mendefinisikan arsitektur dari teknologi *MPLS* terdiri dari gabungan mekanisme konsep *layer 3* (*Routing*) dan *layer 2* (*Switching*). Pada penerapannya paket-paket dari *MPLS* di *forward* dengan menggunakan *routing IGP* dan *EGP*. Dengan memperhatikan sistem *OSI layer*, maka untuk protokol *routing* yang digunakan berada pada *layer* 3 dan *MPLS* berada di antara *layer* 2 dan 3 (Bensalah, El Kamoun, & Bahnasse, 2017).

a.) Arsitektur *MPLS*

Jaringan *MPLS* terbagi atas sirkit yang disebut LSP (*Label-Switced Path*) yang menghubungkan setiap titik yang disebut LSR (*Label-Switched Router*). Setiap LSP dihubungkan dengan sebuah FEC (*forwarding equivalence class*) yang diidentifikasi dengan pemasangan label dan merupakan sekumpulan paket-paket yang menerima aktivitas *forwarding* yang sama di LSR tersebut. Menerapkan LSP memerlukan protokol persinyalan, protokol ini menentukan *forwarding* berdasarkan label pada paket. Dengan label yang ringkas dan berukuran *static* dapat mempercepat proses *forwarding* dan meningkatkan fleksibilitas pemitihan *path*-nya. Sehingga hasilnya *network datagram* menjadi bersifat lebih *connection-oriented* (Bensalah, El Kamoun, & Bahnasse, 2017).

b.) Arsitektur jaringan *MPLS*

Struktur jaringan *MPLS* terdiri dari *edge Label Switching Routers* (ELSR) atau Edge LSRs yang membentangi sebuah *core LSRs*. Berikut elemen-elemen dasar dari penyusunan jaringan *MPLS* antara lain:

- *Edge Label Switching Routers (ELSRs)*

  ELSR terletak pada perbtasan jaringan *MPLS*, yang berfungsi untuk mengaplikasikan label ke dalam paket-paket yang masuk ke dalam jaringan *MPLS*. *MPLS edge router* akan menganalisis *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* (Bensalah, El Kamoun, & Bahnasse, 2017).

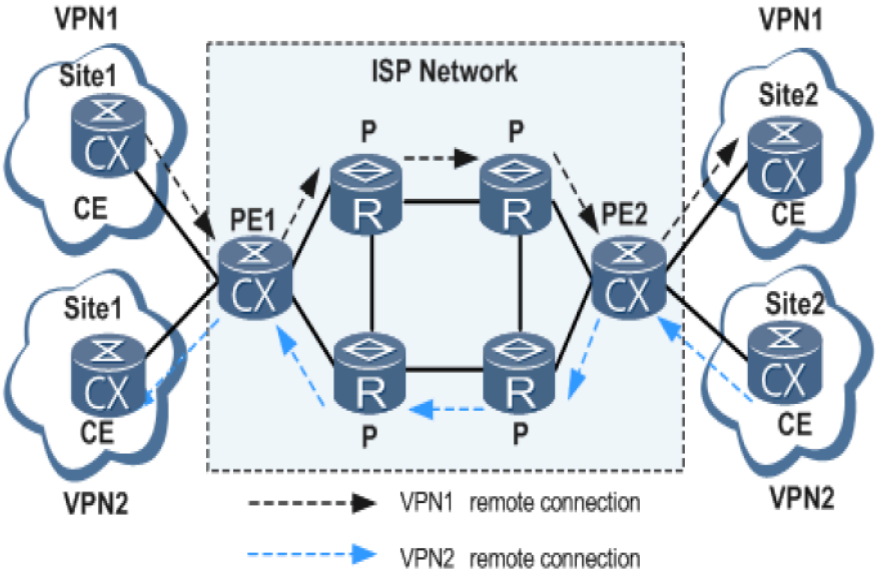
- *Label Distribution Protocol (LDP)*

  LDP merupakan suatu prosedur yang digunakan untuk menginformasikan ikatan label yang telah dibuat satu LSR ke LSR lainnya dalam 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* (Bensalah, El Kamoun, & Bahnasse, 2017).

c.) *MPLS VPN*

Dalam *MPLS VPN* atau singkatan VPLS (Sun & Wu, 2012), data tidak melintasi melalui *internet* tetapi melalui jaringan *MPLS backbone* dari *service provider*. Data-data ketika menerapkan *MPLS VPN* tidak akan diproses di *gateway source* dan *destination*, namun diproses di jaringan *MPLS service provider*. Pada awal paket-paket yang masuk di Label Edge *Router* (LER) diberikan sebuah label yang sesuai dengan sumber dan *tranport mode*-nya., kemudian LER *router* memberikan jalur khusus sesuai masing-masing label. Paket data yang belabel tersebut mengikuti rute-rute

spesifik dan telah diberi keterangan oleh LSR, sehingga paket dapat diarahkan ke jalur yang tepat (Bensalah, El Kamoun, & Bahnasse, 2017).




Gambar 2. 4 *MPLS VPN* (sumber *Evaluation of tunnel layer impact on* VOIP *performances* (IP–MPLS–MPLS VPN–MPLS VPN IPsec)

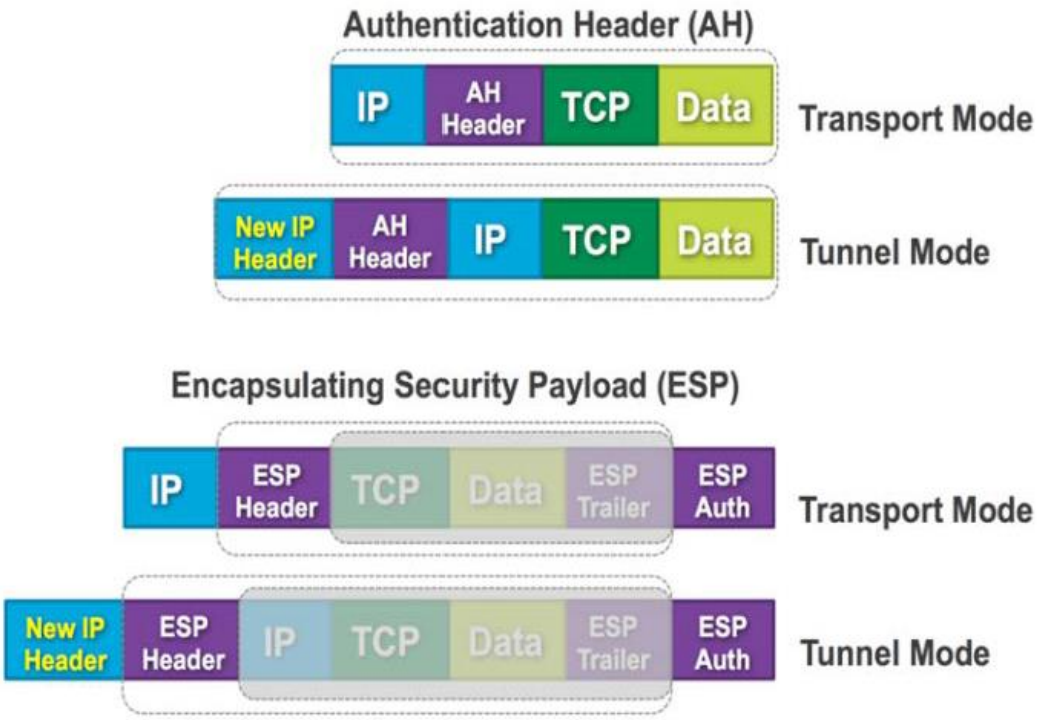
Terdapat dua skenario mengamankan *MPLS* yang dapat diimplementasi sebagai berikut:

- *End-to-end security*: pengguna dapat mengaplikasikan lapisan keamanan yang lebih dengan menggunakan protokol *IPsec* di *Customer edge gateway*. Hal ini tidak disarankan jika pengguna atau perusahaan telah memiliki *service agreement* dengan ISP, karena mengenkripsi *end-to-end* membuat proses klarifikasi menjadi mustahil yang mana mewakili fase awal dari implementasi *QoS policy*.
- *Security between Customer Edge and Provider Edge*: skenario ini paling banyak digunakan karena data dilewatkan tanpa di enkripsi.

d.) *MPLS VPN IPsec*

*IPSec* adalah protokol *tunnel*ing berdasarkan pada dua protokol ESP dan AH. Protokol ESP menjamin kerahasiaan, integritas, dan otentikasi, sedangkan

protokol AH hanya memberikan integritas data dan otentikasi saja (JOKELA, MELEN, & MOSKOWITZ, 2015).



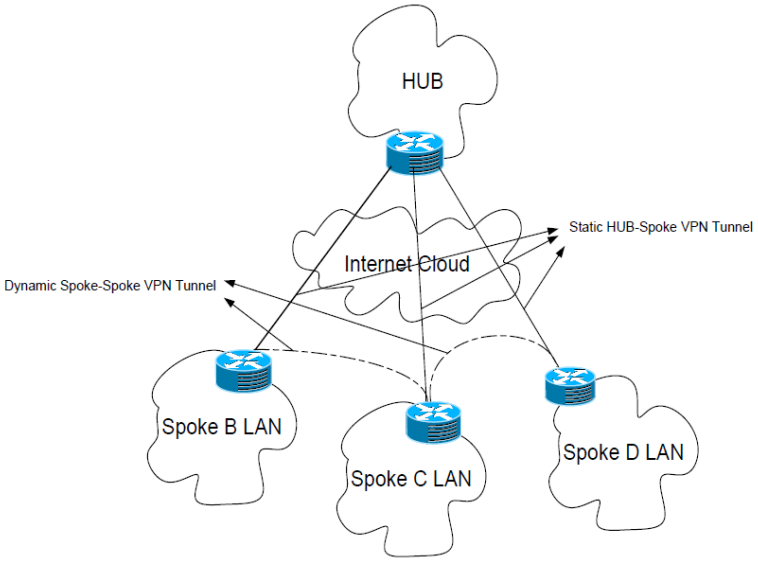

Gambar 2. 5 Metode untuk membangun enkapsulasi *IPsec Header* (sumber *Compression of ipsec ah and esp headers for constrained environments*)

Operasi *IPsec* ada dua cara, pertama *tunnel mode* menambahkan *public IP header* baru dan dienkripsi ke dalam *payload* seperti yang ditunjukkan pada gambar 2.4. Sedangkan, *transport mode* disisipkan antara *network layer* dan *transport layer* dari sistem *OSI model*. Untuk pembasan lebih lanjut akan dirincikan dalam sub-bab 2.2.4 bagian *IPsec ecnryption*. *IPSec* bergantung pada beberapa protokol dalam mempersiapkan fungsionalitasnya, yakni *ISAKMP protocol* yang memanajemen sekumpulan keamanan dan *IKE protocol* untuk menegosiasikan *policy* dan membangun *IPsec tunneling*-nya (RAZA, DUQUENNOY, & SELANDER, 2013).

### 2.2.3 *DMVPN*

*DMVPN* (*Dynamic Multipoint Virtual Private Network*) adalah solusi dari *Cisco IOS Software* dalam membangun skalabilitas *IPsec VPN*. Cisco *DMVPN*

menggunakan arsitektur sentralisasi (*mesh topology*) untuk memberikan kemudahan dalam implementasi dan manajemen terhadap pembangunan *VPN* yang membutuhkan pengaturan verifikasi akses pada grup yang berbeda-beda, seperti *mobile worker*, *telecommuter* dan *extranet user*. Selain itu, memberikan kantor cabang berkomunikasi secara langsung dengan masing-masing kantor cabangnya melalui WAN publik atau *internet*, termasuk penggunaan *voice over IP application* yang tidak memerlukan koneksi *VPN* yang permanen antara *site*. *DMVPN* sendiri memungkinkan *zero-touch deployment IPsec VPN* dan memberikan peningkatan performa jaringan dengan mengurangi *latency* dan *jitter*, sementara mengoptimalkan pemanfaatan *bandwith* kantor pusat (Cisco, Cisco Dynamic Multipoint VPN, 2017).



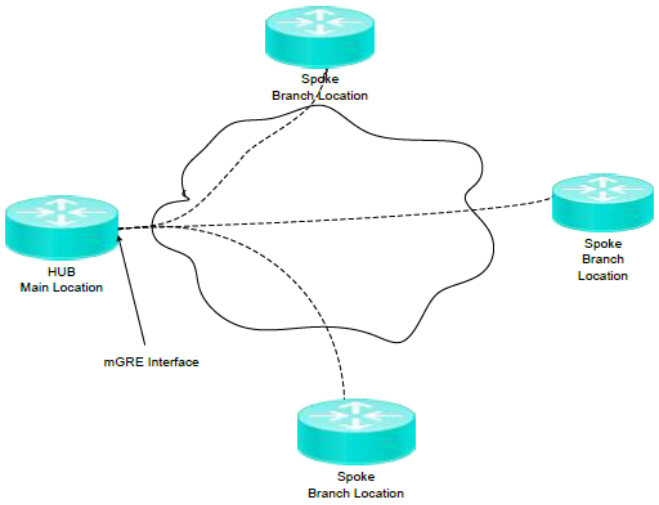

Gambar 2. 6 Model *Dynamic Multipoint VPN Hub-Spoke* (sumber cisco.com)

*DMVPN* merupakan teknologi yang mengintegrasikan konsep berbeda dari protokol dalam meningkatkan keamanan tinggi ketika transmit data melalui jaringan publik. Komponen-komponen tersebut antara lain (Cisco, Cisco Dynamic Multipoint VPN, 2017):

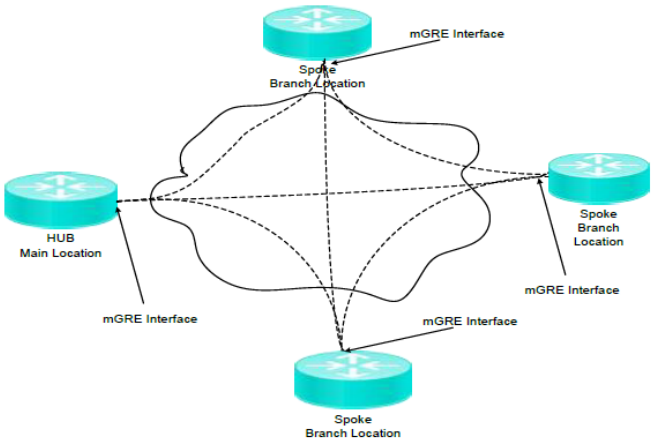
*a.) mGRE* (*Multipoint Generic Routing Encapsulation*)

protokol yang mendukung dalam point-to-point *protocol* dan *static hub and spoke tunnel connectivity*. Keadaan akan menjadi sulit ketika perusahaan memiliki banyak *interface* pada *spoke-to-spoke tunnels* antara lokasi ketika jumlah dari perusahaan/kantor cabang yang semakin meningkat. Sehingga, GRE memberikan bantuan dalam mengurangi jumlah *tunnel interface* dengan mengeliminasi menjadi *single GRE interface*. Berikut keuntungan dari komponen *mGRE*:

- GRE/IPsec *tunnel* dan *endpoints* yang *multiple node* tersebut dapat dijadikan *single GRE interface*.
- *mGRE* mendukung *dynamic tunnel* yang dibangun.
- Dalam pembangunan *VPN*, kompleksitas dari konfigurasi dikurangi secara signifikan
- *mGRE* mendukung *hub and spoke connectivity* melalui *single GRE interface* pada *hub* yang menjadi kantor pusat/*main router*.
- *mGRE* mendukung kantor cabang saling berkomunikasi melalui *single GRE interface*.




Gambar 2. 7 Topologi *Hub and Spoke mGRE tunnel* (sumber cisco)




Gambar 2. 8 Topologi *Spoke to Spoke mGRE tunnel* (sumber cisco.com)

Perbedaan dari *GRE over IPsec* dengan *DMVPN* telah dijabarkan pada tabel 2.1 berikut ini.

Tabel 2. 1 Fitur *GRE over IPsec* dan *DMVPN*

| Features | 3rd party Compatibility | Dynamically Address Spoke | Dynamically Routing | Dynamically Spoke to Spoke *tunnel* | *QoS* |
|---|---|---|---|---|---|
| GRE over IPsec | Yes | - | Yes | - | Yes |
| DMVPN | - | Yes | Yes | Yes | Yes |

| Features | Public Transport | IPv6 | IP Multicast | NAT | VRF |
|---|---|---|---|---|---|
| GRE over IPsec | Yes | Yes | Yes | Yes | Yes |
| DMVPN | Yes | Yes | Yes | Yes | Yes |

(sumber Establishing Secured Enterprise Network Routing protocols by using DMVPN)

*b.)* *NHRP* (*Next-Hop Resolution Protocol*)

Dalam membahas *DMVPN*, diperlukan juga penerapan *NHRP* guna memberikan *client-server model*, yang dimaksud adalah kantor pusat atau *Hub router* sebagai *server* dan kantor cabang atau *Spoke router* menjadi *client*. Hal ini seperti protokol autentikasi yang mengalokasikan sebuah fitur *SHC (Subsequently Hop Client)* dalam mendaftarkan secara cepat terhadap fitur *SHSs (Subsequently Hop Servers)*. Dengan perangkat yang menerapkan *DMVPN*, SHC ini adalah sebuah *spoke router* dan SHS adalah sebuah *hub router* yang sebelumnya seluruh *spoke* telah dibentuk oleh *hub* sehingga, *spoke router* tersebut dapat menentukan *spoke* lain secara dinamikal dengan melihat *NBMA (Non-Broadcast Multi Access) network* yang sama. Berikut keuntungan dari komponen *NHRP* (Kakulapati & Sandhya, Establishing Secured Enterprise Network, 2018):

- *NHRP* memasukkan *NBMA network* pada *ATM* dan standarnya.
- *NHRP* membuat *database* dari *VPN tunnel interface* ke *Internet interface address*.
- *NHRP* mendukung *IPv6* dan memerlukan *IPv6 Unicast Global address* pada masing-masing *interface tunnel*.
- Ketika tidak ada protokol *routing* yang digunakan antara masing-masing *spoke*, tabel *routing* akan di perbaharui dengan *NHRP* "*route*".
- Dalam model *Hub Redudancy*, *IPsec failover* merupakan *statefull* sedangkan *NHRP failover* tidak *statefull*.
- *NHRP* memungkinkan melakukan *QoS policy* pada tiap grup *NHRP* atau per *tunnel* dari *hub* tersebut.
- *NHRP* membatasi fase kedua dari *DMVPN* yang merupakan mengalamatan dari fase ketiga *DMVPN* dengan memiliki

pemerataan *routing* yang baik antara *NHRP route* dan *routing protocol route*.

c.) *IPsec ecnryption – (Optional)*

*IPSec* merupakan mekanisme keamanan *network* layer yang sering gunakan di perusahaan. Hal ini bertujuan untuk memastikan komunikasi yang aman melalui jaringan LAN, WAN dan *Internet*. *IPSec* memberikan keamanan dengan menggunakan dua pengaturan yang berbeda, *AH (Authentication Header)* dan *ESP (Encapsulation Security Payload)*. AH bertugas dalam menentukan autentikasi dan integritas data, sedangkan ESP bertugas memberikan protokol yang terenkripsi dengan integritas sebagai autentikasi opsional terhadap data-data tersebut. Aktivitas dari protokol *IPsec* terdapat dua pendekatan; pendekatan *tunnel* dan pendekatan *transport*, pendekatan *tunnel* merupakan pengembalikan *header IP* dan meringkas paket yang telah selesai, sedangkan pendekatan *transport* tidak merubah header asli dan dimasukkan ke dalam *OSI* model antara *network layer* dan *transport layer* (Kakulapati & Sandhya, Establishing Secured Enterprise Network, 2018).

Dalam *transport mode*, *IPsec* memanfaatkan keaslian dari *IP header* yang sebagai gantinya dari penambahan *tunnel header*. Hal tersebut berjalan dengan baik saat penggunaan *transport mode* dimanfaatkan ketika ukuran paket semakin meningkat. *Transport mode* umumnya digunakan pada koneksi *remote VPN* dengan *software* yang diinstall pada perangkat pengguna yang terhubung ke *VPN server*.

Tabel 2. 2 *Tansport Mode*

| | | | | |
|---|---|---|---|---|
| ESP Auth | ESP Trailer | Payload | ESP Header | Original IP Header |

(sumber *Establishing Secured Enterprise Network Routing protocols by using* DMVPN)

Sementara *tunnel mode*, menggunakan *IP header* yang baru dengan mengenkapsulasikan paket asli dan memiliki *IP source* dan *destination*-nya. *Tunnel mode* ini secara khusus digunakan pada *B2B VPN connection* dimana *source* dan *destination IP* adalah *IP VPN* dari lokasi lainnya.

Tabel 2. 3 *Tunnel Mode*

| | | | | | |
|---|---|---|---|---|---|
| ESP Auth | ESP Trailer | Payload | ESP Header | Original IP Header | New IP Header |

(sumber *Establishing Secured Enterprise Network Routing protocols by using* DMVPN)

d.) *Routing Protocol*

Penukaran informasi tabel *routing* antara *hub* dan *spoke* dari *DMVPN* merupakan tanggun jawab oleh protokol *dynamic routing* karena *router-router* yang menganalisis data dan menemukan rute yang optimal untuk transmisi datanya. Desain *DMVPN* dapat digunakan pada protokol *routing* yang berbeda-beda dan secara teori seluruh *IP based routing* protokol telah didukung. Sementara ketika *NHRP* memfungsikan *routing*-nya, *DMVPN* tetap akan memerlukan protokol *routing* dalam menentukan rute dan memperbaharui informasi tabel *routing*. Pembahasan lebih lanjut mengenai protokol *routing* telah dipaparkan pada sub-bab 2.2.4 (Cisco, Cisco Dynamic Multipoint VPN, 2017).

### *2.2.4* Algoritme kriptografi *Internet Key Exchange (IKE)*

*IKE tunnel* melakukan negosiasi kunci yang akan dipergunakan diantara *peer* serta melakukan negosiasi dalam penentuan algoritme protokol (*AH* dan atau *ESP*) yang akan dipakai. Selain itu, *IKE tunnel* menggunakan algoritme *Diffie-Hellman* untuk menciptakan *key* yang simetris antar *peer* dalam membentuk *tunnel*. Kunci ini hanya akan berlaku disepanjag nilai *time to live*, setelah itu *new key* akan dinegosiasikan kembali. *Tunnel* yang terbentuk disebut *IKE tunnel* atau *Tunnel* negosiasi (Alam, Biddut, Shafin, & Shariar, 2016).

Pada proses menentukan algoritme yang digunakan pada *IKE* meliputi beberapa hal yaitu:

a.) Integritas

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

- *SHA (Secure Hash Standard)*

  *SHA* memproduksi 160-*bit digest*, dimana hasil *hash function* tersebut akan lebih tahan terhadap proses *brute force* daripada *MD5*, tetapi proses ini membutuhkan *resource* yang lebih banyak dibandingkan *MD5*.

- *MD5 (Message Digest)*

  *MD5* memproduksi 128-*bit digest* dimana 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.

b.) Autentikasi

Proses autentikasi dapat dilakukan dengan cara sebagai berikut:

- *Certificate (RSA Encryption)*

  Konfigurasi ini akan memperbolehkan dua *peer* untuk melakukan autentikasi dengan berbagi *public key*, konfigurasi ini dapt melakukan penyangkalan pada negosiasi *IKE*. *RSA Encryption* 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.

c.) Kepastian

- *DES*

  *DES* lebih cepat dibandingkan *3DES* 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)*

  *3DES* bukan merupakan bagian dari *IPsec standard,* karena implementasinya belum dilakukan secara menyeluruh. Kecepatan yang dibutuhkan lebih lambat dan membutuhkan *resource* lebih banyak untuk melakukan tiga kali perhitungan.

d.) Penurunan Kunci

Yaitu pembentukan awal kunci berdasarkan grup *Diffire-Hellman (dh-group)*

- Group 1

  Menggunakan *modulus* 768 (mod768)

- Group 2

  Menggunakan *modulus* 1024 (mod1024)

- Group 5

  Menggunakan *modulus* 1536 (mod1536)

- Group 14

  Menggunakan *modulus* 2048 (mod2048)

- Group 15

  Menggunakan *modulus* 3072 (mod3072)

- Group 16

  Menggunakan *modulus* 4096 (mod4096)

- Group 17

  Menggunakan *modulus* 6144 (mod6144)

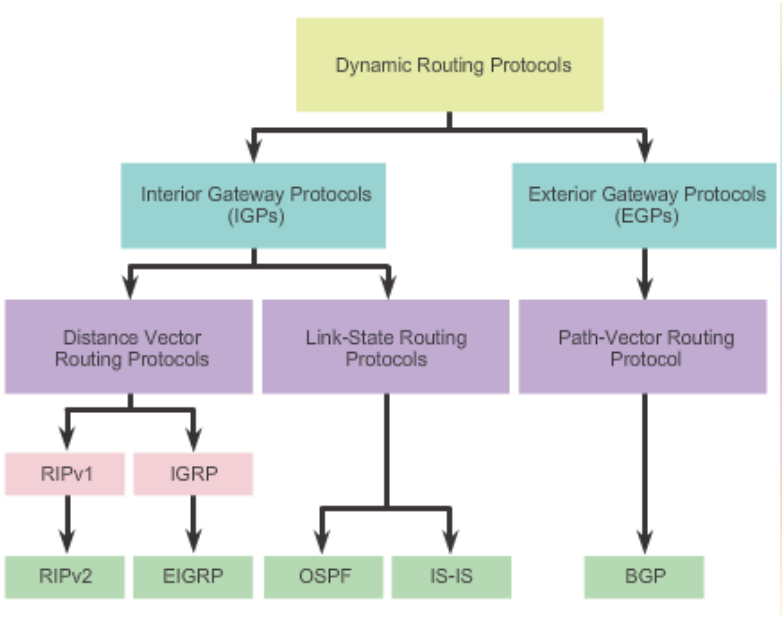
- Group 18
  Menggunakan *modulus* 8192 (mod8192)

### *2.2.5 Routing*

Dalam berkomunikasi antar jaringan, sangat tidak memungkinkan tanpa menggunakan *router* yang menentukan jalur yang terbaik untuk mengirimkan paket dan men-*forward* paket tersebut ke tujuannya. Ketika *router* mendapatkan paket IP dari *interface*, *router* akan menentukan yang mana *interface* digunakan untuk men-*forward* paket ke tujuan berdasarkan tabel *routing*. Informasi *routing* tersebut dibangun dan diperbaharui melalui pertukaran informasi antar *router* dan mempercepat proses perncarian ketika menempatkan rute maupun men-*forward* paket. *Interface router* pada umumnya adalah tujuan akhir dari paket atau sebuah jaringan luar yang terhubung ke *router* lain yang digunakan untuk mencapai tujuan baik jaringan LAN ataupun WAN (Cisco, Cisco Dynamic Multipoint VPN, 2017).

2.2.4.1 *Routing protocol*

Protokol *routing* digunakan untuk menfasilitasi pertukaran informasi routing antara *router-router* dan merupakan serangkaian proses, algoritme dan pesan-pesan yang digunakan dalam pertukaran tersebut, serta memasukkan tabel routing dengan protokol routing-nya untuk memilih rute terbaik. Terdapat dua mekanisme dalam pertukaran informasi routing yakni *static routing* merupakan *routing network* statis yang memberikan kecepatan (tanpa pembaharuan informasi routing) dan kemudahan dalam implementasi. Peracangan *static routing* dapat dikonfigurasi secara manual dan tidak mendukung perubahan jaringan karena pada dasarnya diperuntukkan untuk jaringan skala kecil yang memerlukan sedikit perangkat *router* dan jaringan tidak berkembang secara signifikan. Sedangkan *dynamic routing* dapat memperbaharui informasi routing, menemukan *remote network* atau jaringan diluar cakupan, memilih rute terbaik ke tujuan jaringan dan memiliki kemapuan dalam menemukan rute terbaru jika rute yang sedang digunakan terjadi gangguan (Cisco, Cisco Dynamic Multipoint VPN, 2017).

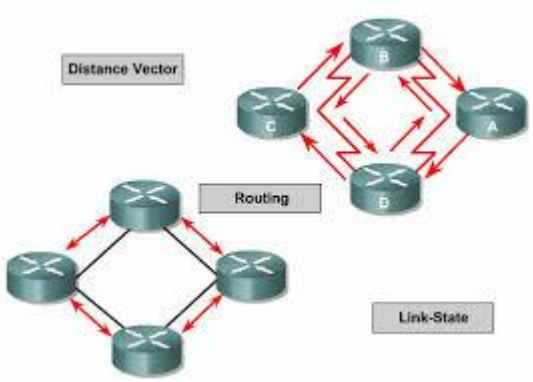



Gambar 2. 9 Protokol *Dynamic Routing* (sumber cisco.com)

Berdasarkan pada karakteristik dari protokol routing yang ditunjukkan pada gambar 2.8, protokol-protokol dapat diklarifikasikan kedalam beberapa grup berbeda (Cisco, Cisco Dynamic Multipoint VPN, 2017) :

*a.) Interior Gateway Protocol (IGP)*

Merupakan *distance vector routing protocol* yang digunakan untuk menghubungkan informasi routing kedalam jaringan *host*. *IGP* memanajemen alur informasi kedalam *router* yang dihubungkan di jaringan host atau *autonomous system*. Protokol memastikan bahwa setiap *router* memiliki tabel routing yang telah diperbaharui dengan menyediakan rute tepat. Selain itu, *IGP* dapat mencegah *routing loops* dari proses perubahan pembaharuan informasi routing yang terjadi pada jaringan ataupun dari kesalahan manajamennya.




Gambar 2. 10 Klarifikasi algoritme *routing* (sumber cisco.com)

Terdapat dua algoritme routing berdasarkan cara pembentukan informasi tabel routing diantaranya (Cisco, Cisco Dynamic Multipoint VPN, 2017) :

- *Distance Vector Routing*

  Protokol *Disctance Vector* berdasarkan pada perhitungan "*Direction*" dan jarak ke beberapa jalur di sebuah jaringan. *Direction* yang dimaksud adalah *next hop address* dan *interface*, yang berfungsi dalam menentukan *cost* dalam menemukan jalur tertentu. *cost* befungsi untuk mencari tujuan dengan menghitung minimal *cost* dari jarak minimal antar dua *link* menggunakan berbagai nilai *route metric*. Protokol routing yang menggunakan *distance vector* adalah *RIP* yang menggunakan *hop count* dari tujuan *forwarding*-nya dimana *EIGRP* membuatnya kedalam informasi lain seperti *node delay* dan *bandwith* yang tersedia.

- *Link State Routing*

  Protokol *Link-state* merupakan salah satu dari dua kelas utama dari protokol routing yang digunakan dalam jaringan *packet switching* untuk komunikasi antar dua *host*. Konsep dasar dari routing ini adalah setiap *node* membangun pemetaan dari konektivitas ke sebuah jaringan, kedalam bentuk grafik dan informasi tersebut menunjukkan *node* yang terhubung ke *node* lainnya serta masing-masing *node* dihitung secara independen pada sebuah *logical path* yang terbaik dari setiap kemungkinan tujuan di sebuah jaringan.

Seluruh informasi rute terbaik dikumpulkan dan dimasukkan ke dalam tabel routing. *OSPF* dan *IS-IS* adalah protokol routing yang menggunakan protokol *link-state*.

*b.) Exterior Gateway Protocol (EGP)*

Merupakan kemampuan protokol dalam mencapai *inter domain* yang digunakan di cakupan *internet*, jaringan internasional, pemerintah, universitas, dan bisnis komersial. *EGP* telah didokumentasikan pada *Request for Comments (RFC) 904*, yang dipublis pada bulan April tahun 1984. Sebagai protokol *exterior gateway* pertama untuk mendapatkan penerimaan yang lebih luas cakupannya di *internet*, *EGP* sendiri dapat menawarkan solusi tersebut. Namun kelemahan dari *EGP* ini menjadikannya lebih transparan atau terlihat di infrastruktur *internet*, sehingga digantikan oleh *exterior gaetway protocol* lain seperti *BGP* dan *Inter Domain Routing Protocol (IDRP)* (Cisco, Cisco Dynamic Multipoint VPN, 2017).

2.2.4.2 *Open Shortest Path First (OSPF)*

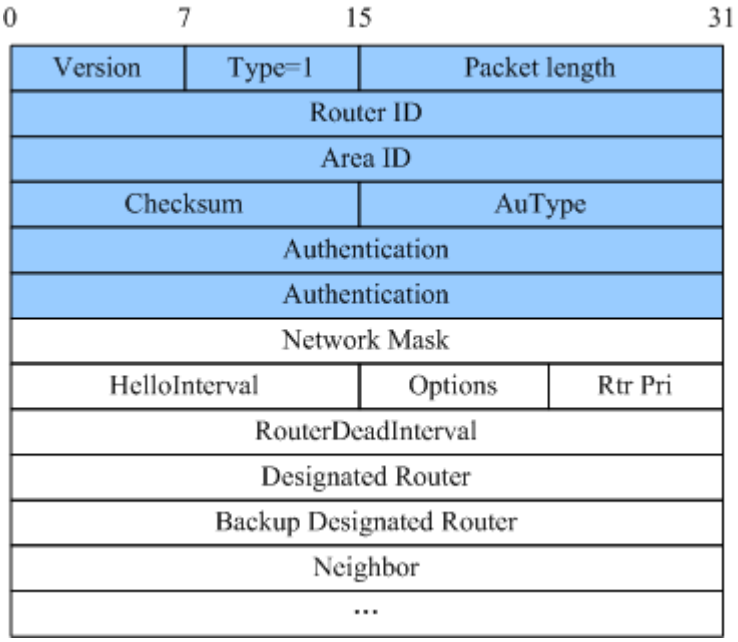
Merupakan sebuah protokol *Gateway Routing Protocol* yang dikembangkan untuk jaringan *berbasis* IP yang berdasarkan algoritme untuk mencari rute tercepat lebih dulu (*shortest path first*) atau *link-state*. *Router* yang menggunakan algoritme *link-state* akan mengirimkan informasi routing ke semua *node* di jaringan dengan menghitung *shortest path* ke setiap *node* berbasis pada topografi dari *Internet* yang dibentuk oleh setiap *node*. Setiap *node* akan mengirimkan bagian dari tabel routing (sambil terus me-*monitor* route ke tujuan jaringan tertentu) yang menjelaskan tentang status dari sambungan dia, *router* juga akan mengirimkan struktur routing (topografi) yang Iengkap (Cisco, Cisco Dynamic Multipoint VPN, 2017).

Keuntungan menggunakan algoritme *shortest path* akan menghasilkan *update* kecil-kecil yang lebih sering. Tabel routing akan lebih cepat konvergen. Hal itu akan menghindari masalah *looping routing*, *count-to-infinity* (pada saat *router* terus-menerus menaikkan *hop count* ke arah sebuah jaringan). Hal-hal ini akan

menyebabkan jaringan yang tidak stabil. Keburukan dari algoritme *shortest path first* membutuhkan *memory* dan kekuatan *CPU* yang besar. Walaupun pada akhirnya lebih banyak keuntungan yang diperoleh dari pengorbanan yang harus dilakukan. *OSPF* versi 2, yang diterangkan di *RFC 1583* (Cisco, Cisco Dynamic Multipoint VPN, 2017).

a.) Tipe-tipe Paket *OSPF*

Terdapat 5 tipe dari paket *OSPF* yang memiliki format *packet header* yang sama. Panjang dari paket tersebut adalah 24 *bytes*, yang digambarkan pada gambar 2.10 dan penjelasan masing-masing format paket sebagai berikut:




Gambar 2. 11 Paket *header OSPF* (sumber cisco.com)

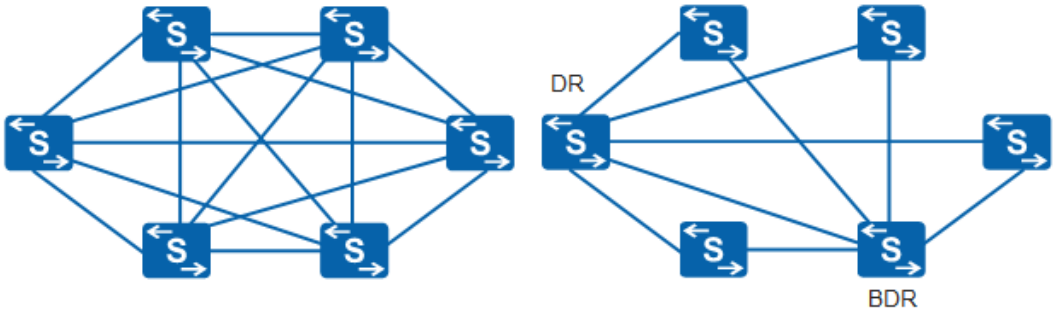
- *Hello packet*

  Pada umumnya digunakan paket yang mana dikirimkan secara berkala oleh *interface* dari *OSPF* untuk membangun dan mempertahankan hubungan/komunikasi dengan *neighbor*. Paket ini berisi informasi terkait *Designated Router, Backup Designated Router, timer* dan informasi *neighbor*.
- *DBD (Database Descriptor Packet)*

Dalam protokol *link state routing*, paket ini menjadi dasar *link state database (LSDB)* untuk mensingkronkan seluruh *router* yang ada. Singkronisasi ini dimulai secepatnya setelah suatu *adjacency* telah terbentuk di antara sebuah *neighbor* dan *OSPF* menggunakan *DBD* sebagai tujuannya. *Database* tersebut dapat dideskripsikan menggunakan paket-paket ganda.

- *LSR (Link State Request)*
  Setelah dua *router* bertukar paket *DBD*, masing-masing tersebut mengirimkan paket *LSR* untuk me-*request* paket *LSA* lain dari *neighbor*-nya. Paket tersebut berisi ringkasan dari *request* dari *LSA*.
- *LSU (Link State Update)*
  Sebuah *router* menggunakan paket *LSU* untuk mengirimkan *LSA* yang di-*request* oleh *neighbor* atau untuk me-*flood* perbaharuan *LSA*. Paket ini berisi pengaturan *LSA* yang tujuan untuk *multicast* dan *broadcast* jaringan.
- *LSAck (Link State Acknowledgement)*
  Sebuah *router* menggunakan paket *LSAck* untuk menyatakan bahwa *LSA* terkandung dalam paket *LSU* yang diterima. *LSA* dapat dinyatakan menggunakan *LSA header* dan paket *LSAck* dapat juga dikirimkan melalui jalur berbeda yakni *unicast mode* ataupun *multicast mode*.

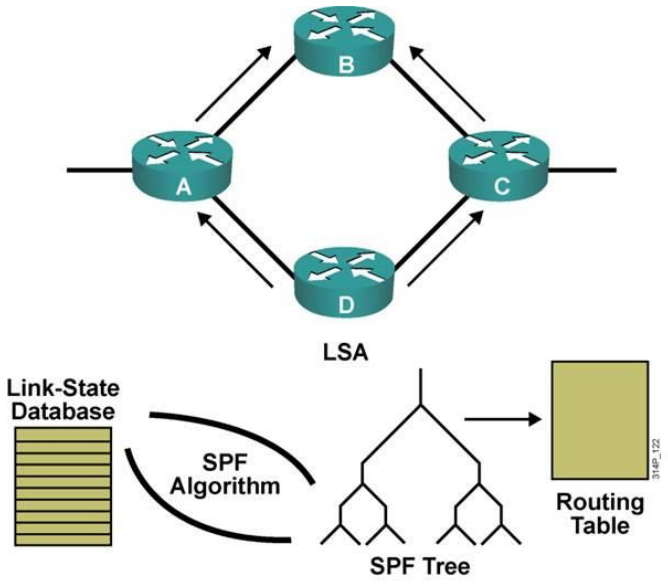
b.) *Designated Router (DR)* dan *Backup Designated Router (BDR)*
Dalam meminimalisirkan jumlah trafik di *multi-access* jaringan *OSPF*, *OSPF* melakukan pemilihan *DR* dan *BDR*. *Router* yang tidak menjadi *DR* atau *BDR* disebut sebagai *DROTHERS*. *DR* bertanggung jawab dalam memperbaharui seluruh *DROTHERS* ketika terjadi perubahan pada jaringan. *BDR* akan melakukan *monitoring DR* dan menggantikan tugasnya ketika *router DR* mengalami kegagalan fungsi atau *fail*. Pada gambar 2.11 menunjukkan sebelum dana sesudah terjadi pemilihan *DR*, *BDR* dan *DROTHERS*.




Gambar 2. 12 Sebelum dan sesudah pemilihan DR dan BDR pada *OSPF* (sumber support.huawei.com)

c.) Algoritme *OSPF*

Algoritme ini ditemukan oleh *Dijkstra* dan menjadi nama dari algoritme *OSPF* yang berfungsi menentukan rute dalam mencapai tujuan paket. Semua *router* dalam area *OSPF* menjalankan algoritme ini secara paralel dan menyimpan hasil ke dalam *link-state database*. Ketika *router* menerima seluruh *LSA* dan membuat *local link state database*-nya, *OSPF* melakukan algoritme ini yang disebut *Dijkstra Shortest Path First* dalam membangun *SPF tree*. *SPF tree* kemudian digunakan dalam membangun tabel routing yang berisi daftar rute atau jalur yang terbaik menuju setiap jaringan.




Gambar 2. 13 Algoritme *OSPF* (sumber support.huawei.com)

2.2.4.3 *Enhanced Interior Gateway Routing Protocol (EIGRP)*

Merupakan evolusi dari *IGRP* yang berusaha memenuhi kebutuhan jaringan skala besar dan perubahan di teknologi jaringan sejak diimplementasikannya *IGRP*. *Router* yang menggunakan *IGRP* dapat menggunakan *EIGRP* karena metrik yang digunakan oleh kedua protokol ini dapat saling translasi. Jika tidak ada *route* yang baik, *EIGRP* akan meminta pada tetangga untuk mencari alternatif *route*. Permintaan ini akan dipropagasikan ke *router* yang lain sampai diperoleh *route* alternatif. Berbeda dengan *IGRP*, *EIGRP* menggunakan *Diffusing-Update Algorithm (DUAL)* yang dikembangkan di *SRI International* (Cisco, Cisco Dynamic Multipoint VPN, 2017).

a.) Tipe-tipe Paket *EIGRP*

*EIGRP* menggunakan 5 tipe paket dalam komunikasi dengan *neighbor*-nya. Tipe paket-paket tersebut diantaranya:

- *Hello Packet*

*EIGRP* menggunakan *hello packet* dalam menemukan *neighbor*-nya. Paket-paket ini mengirimkan melalui *multicast* ke alamat 224.0.0.10 dan secara *default*, *EIGRP* mengirimkan *hello packet* setiap 5 detik.

- *Acknowledgment*

  Paket *acknowledgment* akan menjawab sebuah *update packet* yang diterimanya. Paket tersebut adalah *hello packet* dengan data yang kosong. *EIGRP* mengirimkan *acknowledgment packet* melalui alamat *unicast* dari pengirim paket *update* tersebut.

- *Update*

  *Update packet* berisi informasi routing tujuan paket-paket dan *unicast EIGRP* memperbaharui paket ke *neighbor* yang baru, jika tidak maka *EIGRP* akan melakukan *multicast* sebuah *update packet* ke alamat 224.0.0.10 ketika jalur atau *metric* mengalami perubahan. *Update packet* tersebut merupakan paket yang diakui untuk menentukan transmisi yang andal.

- *Query*

  *EIGRP* mengirim *query packet* untuk menemukan *feasible successors (FS)* ke tujuan secara *multicast*. Ketika *router EIGRP* telah kehilangan informasi terkait jaringan yang ada dan tidak memiliki jalur cadangan maka, *query packet* digunakan untuk menanggulangi masalah tersebut.

- *Reply*

  Paket yang digunakan dalam merespon kepada *query packet* dan memberikan informasi sebuah *FS* ke *router* pengirim secara *unicast*.

b.) Mekanisme *EIGRP*

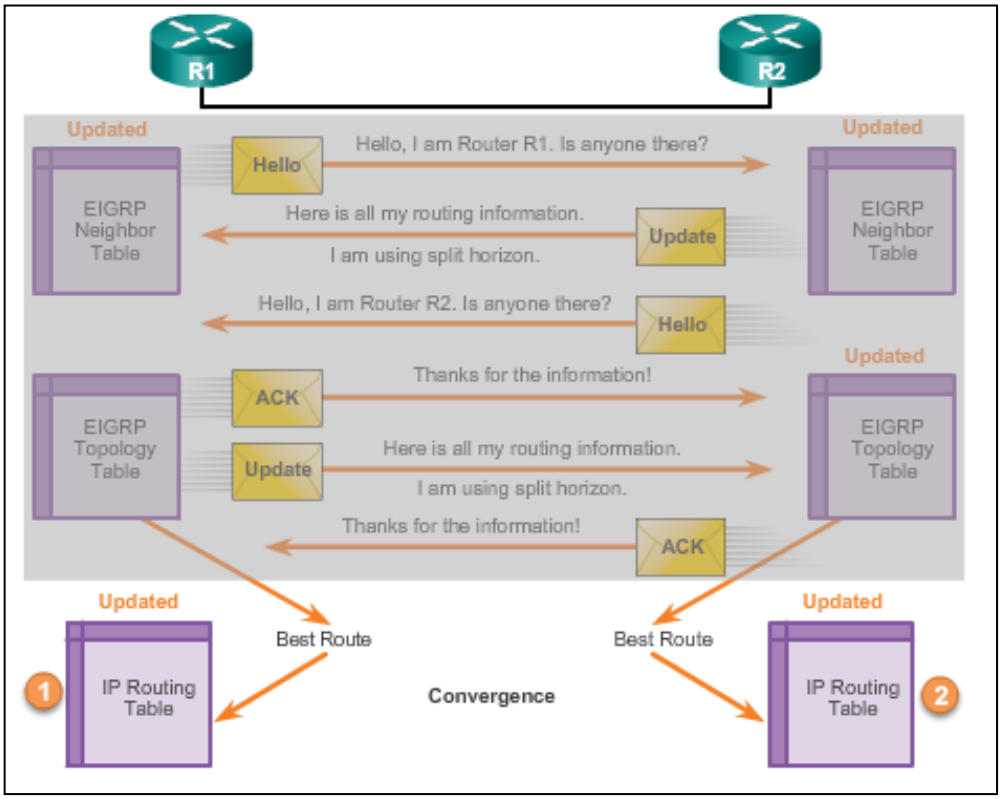
Sebelum *router-router EIGRP* mempersiapkan dalam melakukan pertukaran informasi *route-route* satu dengan yang lainnya, *router* tersebut terlebih dahulu menjadi *neighbor*. Terdapat 3 kondisi yang

diperlukan dalam menetapkan apakah *router* akan menjadi *neighbor* atau tidak. Berikut syarat dalam membangun *neighborship*:

- Menerima paket *Hello* atau *Ack*
- Nomor *Autonomous System* yang sama
- *Metric* yang identik (Nilai K)

Protokol *link state* cenderung menggunakan paket *Hello* dalam menetapkan *neighbor*-nya karena protokol tersebut ketika dalam keadaan normal maka tidak akan mengirimkan informasi terbaru dari *route* dan ke *router* lain dan oleh karena itu, diperlukan sebuah mekanisme dala membantu *neighbor router* untuk mengetahui bahwa ada *router* baru yang telah dipasang ke dalam *environment* jaringan atau *router* lama yang telah *shutdown*. Dalam memperbaharui informasi *neighbor*, *router EIGRP* harus selalu menerima paket *Hello* dari *neighbor router* tersebut.

*Router-router EIGRP* yang memiliki *Autonomous System* yang berbeda-beda maka, secara otomotis pertukaran informasi routing tidak menjadikan *neighbor*. Peristiwa ini dapat menjadi sebuah keuntungan nyata terhadap jaringan berskala luas atau WAN yang bertujuan untuk meminimalisirkan sejumlah informasi *route* yang di *broadcast* melalui nomor *AS* tertentu. Salah satu ketika *EIGRP* melakukan *broadcast* sebuah tabel routing-nya adalah informasi routing akan menemukan *neighbor* baru dan membentuk *adjacency* dengan *neighbor* baru tersebut melalui pertukarang paket-paket *Hello* dan proses ini ditunjukkan pada gambar 2.13.

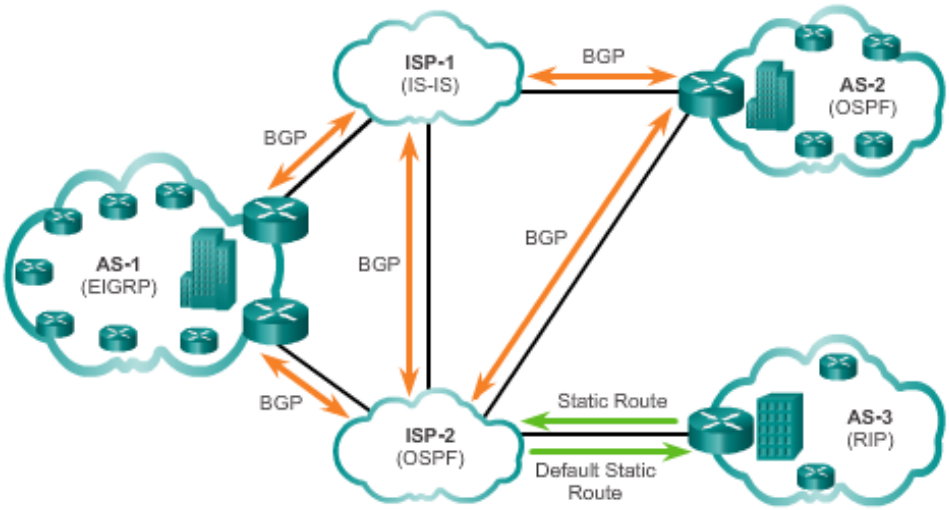



Gambar 2. 14 Proses memperbaharui tabel routing *EIGRP* (sumber cisco.com)

Saat informasi routing terbaru telah diperbaharui dan *router* baru menjadi *neighbor*, maka kedua dari *router* akan melakukan *broadcast* tabel routing-nya kepada *router* lainnya. Setelah seluruh *router* telah mendapatkan informasi dan memperbaharui serta mempelajari pembaharuan tersebut, selanjutnya *router-router* hanya melakukan perubahan informasi pada tabel routing masing-masing yang akan dikirimkan ke *neighbor*-nya. Pada akhirnya, *router EIGRP* telah menerima informasi baru dari tabel routing *neighbor*-nya, *router* tersebut akan menyimpan ke dalam tabel topologi lokal. Tabel ini berisi seluruh *route* yang diketahui dari masing-masing *router neighbor*-nya yang dikenal dan melakukan algoritme sebagai sumber dalam pemilihan jalur terbaik dan di tempatkan ke dalam tabel routing.

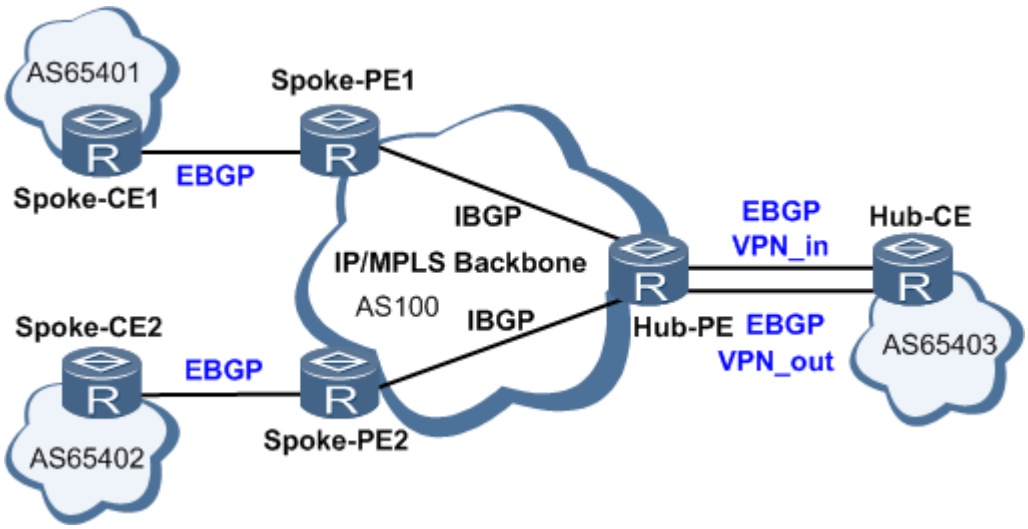
2.2.4.4 *Border Gateway Protocol (BGP)*

*BGPv4* atau *Border Gateway Protocol* versi 4 adalah *exterior gateway protocol* yang memungkinkan sekumpulan *router* dikenal sebagai *autonomous system* untuk saling berbagi informasi routing. *BGP* menggunakan protokol *distance vector*, seperti *RIP*, tetapi tidak seperti protokol *distance vector* yang lain, tabel *BGP* juga menyimpan informasi *route* yang sebenarnya ke jaringan yang dituju. *BGP* juga mendukung kebijakan routing, yang memungkinkan *network administrator* untuk mengatur routing berdasarkan masalah keamanan, politik, legal, atau ekonomis bukan hanya sekadar berdasarkan teknis saja. *BGP* juga mendukung *CIDR* dan *BGPv4* dijelaskan di *RFC 1771*. Sedangkan menurut *RFC 1268* menjelaskan penggunaan *BGP* di *Internet* atau WAN (Cisco, Cisco Dynamic Multipoint VPN, 2017).




Gambar 2. 15 Protokol *Routing* IGP dan EGP (sumber ciscopress.com)

*Autonomous System (AS)* adalah sebuah group dari jaringan yang beroperasi di bawah administrasi bersama, yang menggunakan metodologi routing yang sama. Sebuah *AS* akan menggunakan *Internal Gateway Protocol (IGP)* dan metrik yang sama untuk membuat *route* paket dalam *AS*. *AS* akan menggunakan *External Gateway Protocol (EGP)* untuk membuat *route* paket ke *Autonomous System (AS)* yang lain. Pada sebuah studi penelitian ini, analogi penerapan routing *BGP* pada jaringan *MPLS VPN* digambarkan pada gambar 2.14 dibawah ini (Cisco, Cisco Dynamic Multipoint VPN, 2017).




Gambar 2. 16 Routing pada *MPLS VPN* (sumber ciscopress.com)

### 2.2.6 Video *streaming*

Video *streaming* adalah penggunaan peralatan audio dan video untuk menyelenggarakan konferensi dengan orang-orang yang berada pada lokasi berbeda. Sistem pelayanan ini sekarang masih digunakan hanya untuk tingkat yang masih terbatas. Para pengguna saat ini adalah sektor-sektor bisnis dan industri seperti institusi finansial. Telekomunikasi Video *streaming* menggunakan video dan suara untuk membawa orang pada lokasi berbeda secara bersamaan untuk suatu pertemuan. Ini bisa sesederhana seperti suatu percakapan antara dua orang pada *private offices* (*point-to-point*) atau melibatkan beberapa lokasi (*multi-point*) dengan lebih dari satu orang di dalam ruangan yang besar pada lokasi berbeda. Di samping audio dan *visual transmission*, Video *streaming* dapat digunakan untuk *share* dokumen, informasi *computer-displayed*, dan *whiteboards*.

Teknologi yang digunakan dalam sistem Video *streaming* adalah kompresi digital dari audio dan video *stream*. *Hardware* atau *software* melakukan kompresi yang dinamakan *codec*. Sebuah nomor dari kompresi yang didapatkan sampai 1:500. Hasil *stream* digital dari 1 dan 0 adalah sub bagian dalam paket yang telah dilabelkan, yang mana setelah dilabelkan paket tersebut dikirim melalui jaringan.

Berdasarkan pada tipe koneksi Video *streaming* dibagi menjadi 3 bagian diantaranya:

a.) *Real Time Colaboration Multiparty Conferencing*, komunikasi konferensi yang merupakan teknologi tercepat dengan resolusi yang baik dan interaktif.
b.) *Active Participation partnerships Users*, hubungan yang terjadi antara pengguna dan jaringan komputer atau *database* serta sebuah komunikasi konferensi dengan resolusi baik dan interaktif saat ini.
c.) *Passive Participation partnerships Users*, seorang partisipan, terlibat secara pasif dan memerlukan konferensi dengan komunikasi yang interaktif.

### *2.2.7 International Telecommunication Union (ITU)*

Merupakan sebuah organisasi internasional untuk membakukan atau memastikan dan meregulasi radio internasional dan telekomunikasi, baik di bidang layanan, media dan jaringan yang dipakai, sehingga sebuah jalinan telekomunikasi dapat berjalan lancar. Dengan tujuan untuk membuat standarisasi, pengalokasian spektrum radio, dan mengorganisasikan perjanjian rangkaian interkoneksi antara negara-negara berbeda untuk memungkinkan panggilan telepon internasional. *ITU (International Telecommunication Union)* sendiri merupakan bentukan dari perwakilan pemerintah Eropa pada tahun 1865 dan berdirilah *ITU* di Paris pada tanggal 17 Mei (Quality of Service Regulation Manual, 2017). Ketua *ITU* adalah Houlin Zhao sejak tahun 2014 hingga sekarang, yang merupakan jabatan pada periode ke-19.

Pada paramater yang digunakan mengacu pada standarisasi internasional yaitu *ITU Telecommunication Standardization Sector* (ITU-T*)* yang merupakan salah satu dari tiga sektor unit dari *International Telecommunication Union* (ITU).

#### *2.2.7.1 Throughput*

*Throughput* merupakan jumlah total kedatangan paket yang sukses diamati pada *destination* selama interval waktu tertentu dibagi oleh durasi interval waktu tersebut. *Throughput* merupakan total jumlah *bit* paket selama transfer dibagi dengan durasi selang waktu transfer tersebut.

#### *2.2.7.2 Delay*

*Delay* merupakan waktu yang dibutuhkan oleh sebuah paket data terhitung dari saat pengiriman oleh *transmitter* sampai saat diterima oleh *receiver*. *Delay* menjadi acuan waktu transmisi paket dari pengirim hingga ke penerima. Berikut merupakan standar *delay* menurut *ITU-T G.114*.

Tabel 2. 4 Standarisasi kualitas *Delay*

| Category | *Delay* |
|---|---|
| Excellent | < 150 ms |
| Good | 150 s/d 300 ms |
| Poor | 300 ms s/d 450 ms |
| Bad | > 450 ms |

#### *2.2.7.3 Jitter*

*Jitter* adalah variasi *delay*, yaitu perbedaan selang waktu, *Jitter* merupakan variasi *delay* antar paket yang terjadi pada jaringan IP. *Jitter* dapat disebabkan oleh terjadinya kongasti, kurangnya kapasitas jaringan, variasi ukuran paket serta ketidak urutan paket. Berikut merupakan standar *jitter* menurut *ITU-T G.114*.

Tabel 2. 5 Standarisasi kualitas *Jitter*

| Category | *Jitter* |
|---|---|
| Good | 0-20 ms |
| Fair | 20-50 ms |
| Bad | > 50 ms |

#### *2.2.7.4 Packet Loss*

*Packet loss* merupakan penyebab utama pelemahan transfer *VoIP*. *Packet loss* terjadi karena pembuangan paket dalam jaringan (network loss) atau pembuangan paket di *gateway* samapai kedatangan terakhir (late loss). Berikut merupakan standar *packet loss* menurut *ITU-T G.114*.

Tabel 2. 6 Standarisasi kualitas *Packet loss*

| Quality | *Packet loss* |
|---|---|
| Excellent | 0 - 0.5% |
| Good | 0.5 - 1.5% |
| Bad | > 1.5% |