BAB 2
DASAR TEORI

2.1 KAJIAN PUSTAKA

Penelitian yang dilakukan Akhmad Zulfikar Al Ghivani pada tahun 2018 [1] yang berjudul "Studi Perbandingan *Routing* Protokol BGP Dan EIGRP, Evaluasi Kinerja Performansi Pada *Autonomous System* Berbeda". Penelitian ini membandingkan data QoS (*throughput* dan *delay*) antara protokol BGP dan EIGRP dimana setiap topologi masing-masing *router* memiliki *Autonomous System*. Hasil dari penelitian ini adalah protokol EIGRP lebih baik dibandingkan dengan protokol BGP.dimana nilai *delay* yang didapat lebih kecil dan nilai *throughput* yang di dapat lebih besar pada EIGRP daripada protokol BGP, yaitu 0,013333 detik untuk *delay* dan 2378,307 bps untuk *throughput*.

Tahun 2019 Muhammad Ikhsan Azhari, Tengku Mohd. Diansyah, Ari Usman [2] melakukan penelitian yang berjudul "Perbandingan *Routing protocol Exterior* BGP Versi 4 Dengan *Routing Interior* EIGRP Pada Algoritma *Linkstate* Menggunakan Parameter *Packet loss*". Penelitian ini berisi tentang pemilihan jalur *routing* terbaik antara protokol BGP versi 4 dan protokol EIGRP dengan parameter yang dibandingkan adalah *packetloss*. Berdasarkan penelitian tersebut protokol EIGRP dan BGP memiliki *packet loss* 0% yang berarti pengiriman paket sukses. Namun protokol EIGRP lebih cepat dan efisien dibandingkan protokol BGP dalam pengonfigurasian dan transfer data.

Penelitian yang dilakukan Dudy Supriadi, Andy Hidayat Jatmika, I Wayan Agus Arimbawa [3] pada tahun 2019 yang berjudul "Analisis Perbandingan Protokol *Routing* OSPF dan RIPv2 Berdasarkan Variasi Jumlah *Router* Pada Jaringan MPLS dan Tanpa MPLS Menggunakan Simulator GNS3" membahas tentang kinerja protokol *routing* OSPF dan RIPv2 pada jaringan MPLS dan non-MPLS dengan memvariasikan *router* masing-masing 5, 10, dan 15. Hasil dari penelitian tersebut adalah adanya peningkatan kinerja protokol *routing* OSPF dan RIPv2 dengan MPLS. Parameter *bitrate* mengalami peningkatan sebesar 36,40%,

jitter mengalami peniongkatan sebesar 15,14%, dan peningkatan *packet loss* sebesar 12%. Pada protokol RIPv2 saat mengimplementasikan MPLS. Di sisi lain, OSPF yang menggunakan MPLS mengalami peningkatan nilai *bitrate* sebesar 36,86%, nilai *jitter* sebesar 20,65%, dan *packet loss* sebesar 8%.

Tahun 2019 Pahlevi Muhammad, Primantara Hari Trisnawan, Kasyful Amron [8] melakukan penelitian yang berjudul "Analisis Perbandingan Kinerja Protokol *Routing* OSPF, RIP, EIGRP, Dan IS-IS" membahas tentang performa *routing* protokol OSPF, RIP, EIGRP, serta IS-IS di IPv4 dan IPv6 dalam mengirimkan sebuah informasi. Pengujian tersebut menggunakan topologi *Mesh* dengan jumlah *router* 4, 6, dan 8 *router*. Parameter yang digunakan pada penelitian ini adalah waktu konfergensi dan waktu *round-trip*. Berdasarkan pengujian tersebut protokol EIGRP dan IS-IS mendapatkan nilai 14 detik pada skenario waktu konvergensi IPv4 dengan 4 *router* dan nilai 16 detik pada 8 *router*, namun pada pengujian 6 *router* protokol RIP dan IS-IS mendapatkan nilai 8 detik. Sedangkan pada skenario waktu konvergensi IPv6 protokol EIGRP mendapatkan nilai 14 detik di pengujian 4 *router* dan 4 detik pada 6 *router*, namun pengujian 8 *router* protokol EIGRP dan IS-IS yang mendapatkan nilai rata-rata terkecil yaitu 16 detik. Pada skenario waktu *round-trip* IPv4 protokol EIGRP mendapatkan nilai rata-rata terkecil pada pengujian 4 *router*, namun pada pengujian 6 dan 8 *router* protokol IS-IS yang mendapatkan nilai rata-rata terkecil yaitu 128 detik dan 146 detik. Sedangkan pada skenario waktu konvergensi IPv6 protokol RIPng mendapatkan nilai 67 detik dan 115 detik pada pengujian 4 dan 6 *router,* serta pada pengujian 8 *router* protokol OSPFv3 mendapatkan nilai 130 detik.

Penelitian yang dilakukan pada tahun 2020 oleh Supriyatno, Jupriyadi, Syaiful Ahdan, dan Sampurna Dadi Riskiono [10] yang berjudul "Analisis Perbandingan Kinerja Protokol *Routing* RIP Dan OSPF Pada Topologi *Mesh*" membahas tentang penentuan metode *routing* dalam jaringan komunikasi data untuk meningkatkan performa jaringan. Penelitian tersebut membandingkan kinerja protokol *routing* RIP dan OSPF berdasarkan QoS meliputi *delay*, *packet loss*, dan *throughput* pada topologi *Mesh*. Berdasarkan penelitian tersebut saat kondisi jaringan normal, protokol OSPF memiliki performa yang lebih baik

dibandingkan protokol RIP. Sedangkan saat kondisi jaringan sibuk, performa protokol OSPF menurun dibandingkan protokol RIP.

Tahun 2017 penelitian berjudul "*Routing protocol RIPng, OSPFv3, and EIGRP on IPv6 for Video streaming Services*" yang dilakukan oleh Nurhayati, Rahmat Fajar Al Farizky [11] membahas tentang semakin meningkatnya pengguna internet yang sebanding dengan meningkatnya kebutuhan IP. Solusi untuk mengatasi keterbatasan IPv4 adalah dengan mengembangkan IPv6. Hasil dari penelitian ini adalah protokol *routing* EIGRP memiliki kriteria QoS yang lebih baik dibandingkan RIPng dan OSPFv3 untuk layanan *video streaming* dengan memiliki nilai rata-rata *delay* 14,2 ms, *packet loss* 2,5%, dan *throughput* 714,1 Kbit/s.

Tabel 2. 1 Kajian Penelitian Sebelumnya

Tahun	Nama	Judul	Hasil
2018	Akhmad Zulfikar Al Ghivani	Studi Perbandingan *Routing* Protokol BGP Dan EIGRP, Evaluasi Kinerja Performansi Pada *Autonomous System* Berbeda	EIGRP lebih unggul dari BGP, karena nilai *throughput* lebih besar serta nilai *delay* yang lebih kecil dibandingkan dengan BGP
2019	Muhammad Ikhsan Azhari, Tengku Mohd. Diansyah, Ari Usman	Perbandingan *Routing protocol Exterior* BGP Versi 4 Dengan *Routing Interior* EIGRP Pada Algoritma *Linkstate* Menggunakan Parameter *Packet loss*	Penggunaan *routing* EIGRP lebih efisien dalam pengkonfigurasian serta kecepatannya lebih baik dibandingkan dengan BGP
2019	Dudy Supriadi, Andy Hidayat Jatmika, I Wayan Agus Arimbawa	Analisis Perbandingan Protokol *Routing* OSPF dan RIPv2 Berdasarkan Variasi Jumlah *Router* Pada Jaringan MPLS dan Tanpa MPLS Menggunakan Simulator GNS3	Terjadi peningkatan kinerja pada protokol *routing* OSPF dan RIPv2 dengan MPLS
2019	Pahlevi Muhammad, Primantara Hari Trisnawan, Kasyful Amron	Analisis Perbandingan Kinerja Protokol *Routing* OSPF, RIP, EIGRP, Dan IS-IS	Nilai yang dihasilkan pada tiap skenario yang diujikan bervariasi.

Tahun	Nama	Judul	Hasil
2020	Supriyatno, Jupriyadi, Syaiful Ahdan, Sampurna Dadi Riskiono	Analisis Perbandingan Kinerja Protokol *Routing* RIP Dan OSPF Pada Topologi *Mesh*	Kinerja *routing* protokol OSPF lebih unggul dari *routing* protokol RIP saat kondisi jaringan tidak sibuk, sedangkan saat kondisi sibuk performa OSPF menurun dibandingkan protokol RIP
2017	Nurhayati, Rahmat Fajar Al Farizky	*Routing protocol RIPng, OSPFv3, and EIGRP on IPv6 for Video streaming Services*	Protokol *routing* EIGRP memiliki kriteria QoS yang lebih baik dibandingkan RIPng dan OSPFv3 untuk layanan *video streaming*.

2.2 JARINGAN KOMPUTER

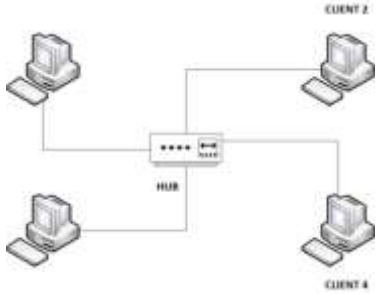
Kumpulan berbagai komputer yang dapat terhubung satu sama lain dan berkomunikasi satu sama lain untuk bekerja sebagai satu unit disebut jaringan komputer. Sebuah jaringan komputer memungkinkan dua atau lebih komputer untuk bertukar data, berkomunikasi, dan menjalankan aplikasi yang sama. Jaringan komputer dan komunikasi dimungkinkan. Ada banyak jenis kabel yang dapat digunakan untuk menghubungkan jaringan komputer, termasuk kabel tembaga, kabel koaksial, kabel twisted pair, kabel serat optik, dan teknologi nirkabel. Semua ini memungkinkan komunikasi komputer ke komputer dan transfer data.

Jaringan komputer dapat mengambil berbagai bentuk, seperti Local Area Network (LAN), yang menjangkau wilayah kecil seperti kantor atau gedung, Metropolitan Area Network (MAN), yang menjangkau wilayah yang lebih besar seperti kota, dan Wide Area Network (WAN), yang menjangkau wilayah yang lebih besar lagi seperti antara kota atau negara. Jaringan komputer memungkinkan akses komunikasi dan informasi secara efisien dan efektif [12].

a. *Local Area Network* (LAN)

Istilah "*Local Area Network*" (LAN) mengacu pada jaringan pribadi yang membentang beberapa kilometer di dalam struktur tunggal. Fungsinya adalah untuk memungkinkan berbagi sumber daya dan pertukaran data antara komputer

desktop dan *workstation* dalam lingkungan perusahaan atau manufaktur. Gambar 2.1 adalah ilustrasi dari *Local Area Network* (LAN).

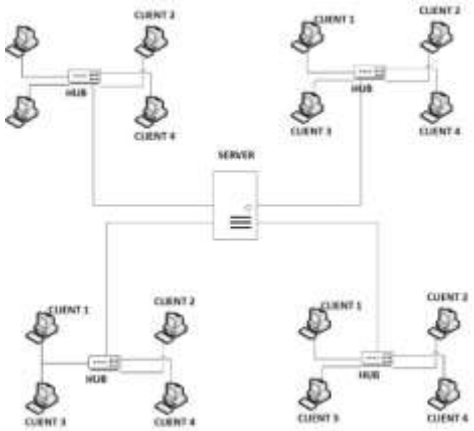


Gambar 2.1 *Local Area Network*

Ketiga karakteristik utama yang membedakan LAN dari jenis jaringan lainnya adalah ukurannya, teknologi transmisi yang digunakan, dan topologinya. LAN sering menghubungkan banyak komputer yang relatif berdekatan, memungkinkan pertukaran data dan berbagi sumber daya yang efisien. Dalam contoh yang diberikan, beberapa PC dapat terhubung melalui sebuah *hub*, yang bertindak sebagai pusat penghubung untuk mengizinkan pertukaran informasi antar komputer yang masih berada dalam jaringan yang sama.

b. *Metropolitan Area Network* (MAN)

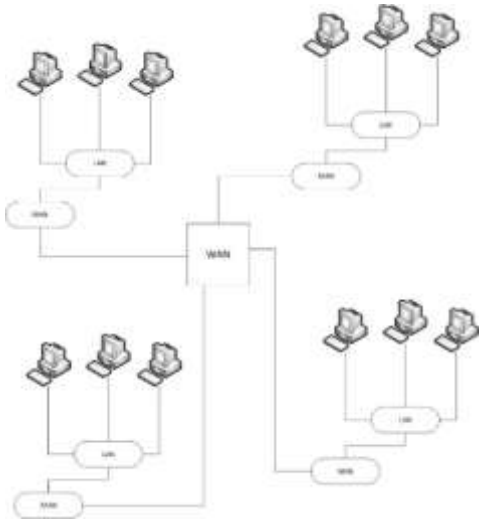
LAN yang disempurnakan yang dikenal sebagai *Metropolitan Area Network* (MAN) biasanya menggunakan teknologi seperti LAN. MAN memiliki cakupan lebih luas daripada LAN dan biasanya mencakup area kota atau wilayah perkotaan. Jaringan ini dapat digunakan untuk keperluan pribadi maupun publik. Jaringan MAN dapat dibentuk dengan menghubungkan beberapa LAN yang terletak pada lokasi-lokasi yang berbeda, membentuk jaringan yang lebih besar dan terintegrasi. Gambar 2.2 merupakan ilustrasi jaringan MAN.



Gambar 2.2 *Metropolitan Area Network*

c. *Wide Area Network* (WAN)

Wide Area Network (WAN) merupakan tipe jaringan komputer yang melibatkan area yang besar dalam cakupannya, termasuk tetapi tidak terbatas pada skala antar kota, antar wilayah, dan bahkan melintasi batas negara. WAN kadang-kadang digunakan untuk merujuk ke jaringan komputer yang membutuhkan *router* dan saluran komunikasi terbuka sehingga data dapat dikirim antar situs. WAN kadang-kadang digunakan untuk merujuk ke jaringan komputer yang membutuhkan *router* dan saluran komunikasi terbuka sehingga data dapat dikirim antar situs. Jaringan WAN sering dilihat sebagai pengelompokan atau kumpulan berbagai jaringan LAN dan MAN yang terpisah, membentuk jaringan yang lebih besar dan terhubung secara global. Dalam gambar 2.3, dapat dilihat contoh ilustrasi jaringan WAN yang menggabungkan beberapa LAN dan MAN untuk membentuk jaringan yang luas dan kompleks [13].



Gambar 2.3 *Wide Area Network*

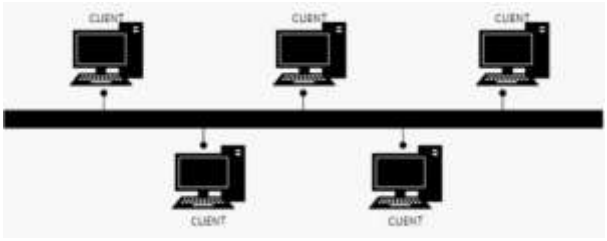
2.3 TOPOLOGI JARINGAN KOMPUTER

Penyusunan rencana koneksi di dalam Jaringan Area Lokal (LAN) yang seringnya mengandalkan kabel sebagai media utama untuk berkomunikasi, melibatkan penggunaan konektor, teknologi ethernet, serta perangkat pendukung lainnya. Konsep ini dikenal sebagai topologi jaringan atau arsitektur jaringan. Berikut adalah beberapa jenis topologi jaringan yang sering digunakan dalam jaringan lokal area.

a. Topologi *Bus*

Topologi *Bus* merupakan suatu konfigurasi jaringan di mana terdapat satu kabel tunggal yang ditutup di kedua ujungnya, dan terdapat simpul atau *node* yang berjarak disepanjang kabel tersebut, ditunjukkan pada gambar 2.4. Topologi ini terjadi saat sinyal hanya dibawa dalam satu arah melintasi kabel, sehingga dapat terjadi tabrakan (*collision*) ketika beberapa perangkat mengirim data secara bersamaan.

Keuntungan dari topologi ini adalah biayanya yang murah karena menggunakan sedikit media transmisi (kabel), dan kabel yang digunakan umumnya mudah ditemukan di pasaran. Selain itu, topologi bus juga memungkinkan adanya koneksi langsung antara masing-masing komputer. Bersamaan dengan keuntungan, topologi *bus* juga memiliki kelemahan, ketika jalur yang sama digunakan oleh lebih dari satu perangkat secara bersamaan, dapat terjadi gangguan atau *crosstalk* yang menyebabkan ketidakstabilan koneksi atau kualitas sinyal yang menurun.



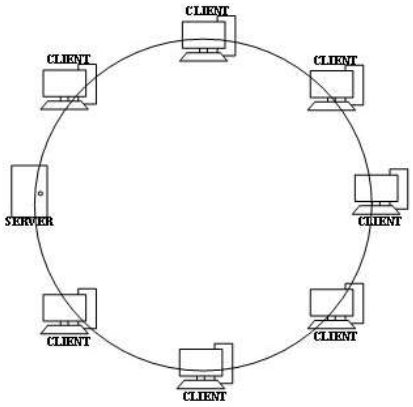
Gambar 2.4 Topologi *Bus*

b. Topologi *Ring*

Topologi *Ring* adalah jenis konfigurasi jaringan yang memiliki simpul dan berbentuk seperti lingkaran tertutup, ditunjukkan pada gambar 2.5. Dalam topologi ini, sinyal data mengalir dalam dua arah untuk mencegah kemungkinan terjadinya tabrakan data yang cepat. Setiap komputer terhubung dalam lingkaran koneksi yang mengelilingi setiap komputer, mirip dengan topologi bus tetapi dengan ujung *bus* disambungkan. Untuk memastikan data yang dikirimkan mencapai komputer yang dituju, setiap komputer diberikan alamat tujuan.

Keuntungan dari menggunakan topologi jaringan ini adalah kemampuannya dalam mengurangi kesalahan transmisi dengan menggunakan koneksi *point-to-point*, dan ketika ada gangguan pada media koneksi, data masih

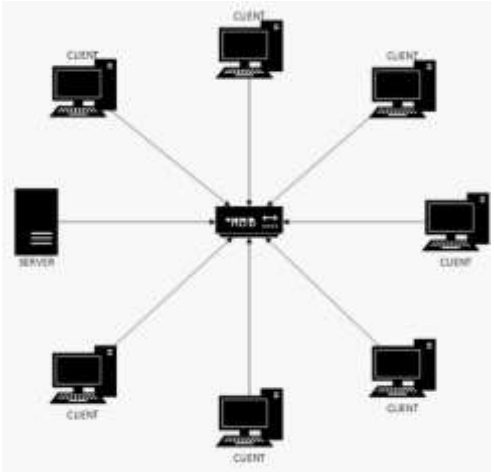
dapat dikirim melalui jalur lain yang masih aktif. Kelemahan topologi *Ring* adalah data harus melewati banyak komputer dalam lingkaran, transfer data dapat menjadi lambat karena harus melalui setiap simpul dalam jaringan.



Gambar 2. 5 Topologi *Ring*

c. Topologi *Star*

Topologi jaringan di mana setiap *node* atau stasiun berkomunikasi secara langsung dengan *hub* atau *switch* disebut topologi *star*. Topologi ini dapat dilihat pada gambar 2.6. Data lalu lintas mengalir dari *node* ke *hub* atau *switch*, dan kemudian dirutekan ke *node* tujuan (stasiun). Jaringan lain tidak akan terputus apabila salah satu segmen kabel tidak terhubung atau putus karena adanya *hub*ungan langsung antara *node* dan *hub*/*switch*.



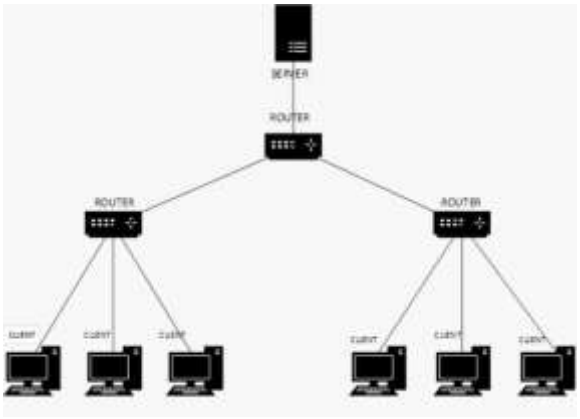
Gambar 2.6 Topologi *Star*

Manfaat menggunakan topologi *star* termasuk penambahan *workstation* baru yang mudah selama *port* pada *hub* / *switch* masih tersedia, akses cepat ke stasiun lain (*client* atau *server*), dan kapasitas untuk mengatur *hub* / *switch* dalam

tingkatan untuk meningkatkan jumlah stasiun yang terhubung ke jaringan. Selain itu, topologi *star* juga memungkinkan untuk menambahkan lebih banyak pengguna daripada topologi *Bus* atau *Ring*. Topologi *star* juga memiliki beberapa kelemahan, jika terjadi tabrakan karena lalu lintas data yang tinggi, semua komunikasi akan tertunda dan jika *hub* atau *switch* menentukan bahwa tidak ada jalur alternatif yang dapat digunakan oleh *node* lain, koneksi akan berlanjut secara acak.

d. Topologi *Tree*

Topologi *Tree* di dalamnya terdapat perbedaan lokasi pada setiap stasiun dalam jaringan. *Hierarchical topology* (topologi hirarkis) digunakan, di mana stasiun di lokasi yang lebih tinggi memiliki kendali atas stasiun di lokasi yang lebih rendah, seperti yang terlihat pada gambar 2.7. Di sisi lain, terdapat juga *peer topology* (topologi sejajar) di mana stasiun-stasiun memiliki posisi yang sama dan saling berkomunikasi secara langsung tanpa.



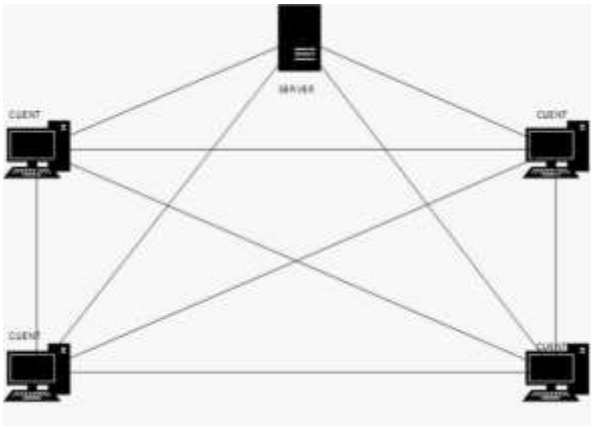
Gambar 2.7 Topologi *Tree*

e. Topologi *Mesh*

Topologi *Mesh*, dalam struktur jaringannya, memiliki koneksi antara sentral (pusat) yang digunakan sepenuhnya. Semakin banyak pusat yang terhubung, semakin kompleks jaringan tersebut, biaya operasionalnya juga meningkat secara linier. Selain itu, struktur ini cenderung kurang ekonomis dan memerlukan biaya operasional yang tinggi [14].

Gambar 2.8 merupakan topologi *mesh*. Topologi *mesh* merupakan jenis konfigurasi jaringan komputer di mana setiap node terhubung langsung ke setiap simpul lainnya dalam jaringan. Dengan kata lain, setiap perangkat memiliki

koneksi langsung dengan semua perangkat lainnya. Topologi ini menciptakan jaringan yang sangat redundan dan tahan terhadap kegagalan, karena jika satu jalur koneksi rusak, masih ada jalur alternatif yang dapat digunakan untuk komunikasi.



Gambar 2.8 Topologi *Mesh.*

Topologi *mesh* dapat dibagi menjadi dua jenis, yaitu *full mesh* dan *partial mesh* atau dapat juga disebut dengan half mesh. Dikatakan *full mesh* dikarenakan setiap nodenya saling terkoneksi antara node yang satu dengan node lainnya. Sedangkan *half mesh* setiap nodenya tidak harus terhubung dengan masing-masing node yang ada melainkan terhubung dengan satu atau beberapa saja [15].

2.4 MPLS (*MULTI PROTOCOL LABEL SWITCHING*)

Multiprotocol Label Switching (MPLS) adalah metode berkinerja tinggi untuk merutekan paket data dalam jaringan. Arsitektur jaringan MPLS, yang menggabungkan teknologi *layer* 2 label *switching* dengan *layer* 3 *routing* untuk mempercepat pengiriman paket, diciptakan oleh *Internet Engineering Task Force* (IETF). MPLS memungkinkan *router* di jaringan tepi untuk langsung menambahkan label ke paket data. Akibatnya, setiap paket data dapat memiliki label yang ditambahkan oleh perangkat MPLS seperti sakelar ATM atau *router* dalam penyedia internet. Karena teknologi ATM dan *frame-relay* berorientasi koneksi, protokol terpisah harus dibuat untuk setiap sirkuit virtual sebelum transmisi. Sebaliknya, protokol *routing* yang digunakan oleh IP memutuskan

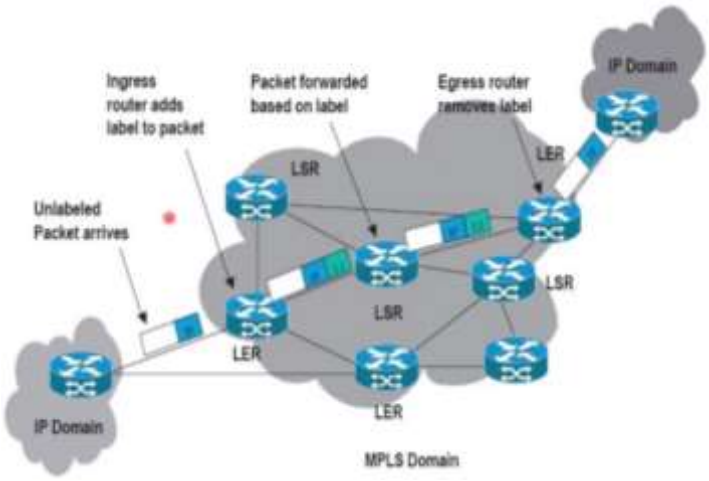
paket *path* yang diambil dengan berbagi informasi *routing* [16]. Komponen MPLS antara lain :

1. *Any Transport over* MPLS (AToM) adalah *protocol* yang membawa traffic pada *layer* 2 untuk melewati *cloud* MPLS.
2. *Asynchronous Transfer Mode* (ATM) adalah teknik transfer data berbasis sel dimana kebutuhan kanal jaringan menentukan alokasi paket.
3. *Forwarding Equivalence Class* (FEC) adalah kumpulan paket yang menerima perlakuan yang sama pada proses pengiriman paket data pada LSR.
4. *Ingress* adalah *router* pertama pada cloud MPLS tempat masuk paket data.
5. *Egress* adalah *router* terakhir pada cloud MPLS tempat keluar paket data.
6. *Label Distribution Protocol* adalah *protocol* yang mengatur pendistribusian paket pada jaringan MPLS.
7. *Label Switching Router* (LSR) adalah node MPLS yang mampu meneruskan paket data pada proses *routing* di *layer* 3.
8. *Label Switching Path* (LSP) adalah jalur yang melewati satu atau serangkaian LSR untuk meneruskan paket dari satu node MPLS ke node MPLS lain.
9. *Label-switching table* adalah *table* yang berisi data label yang diberikan pada paket data.

Jaringan MPLS terbentuk oleh serangkaian node yang memiliki kemampuan untuk beralih (*switch*) serta merutekan (*route*) paket yang telah diberi tanda label. Paket data yang berlalu dalam jaringan MPLS membutuhkan penambahan label pada bagian *header* agar dapat mengidentifikasi tujuan akhirnya. Penambahan atau penghapusan label ini dikelola oleh *Label Edge Router* (LER) ketika paket memasuki jaringan (*ingress*).

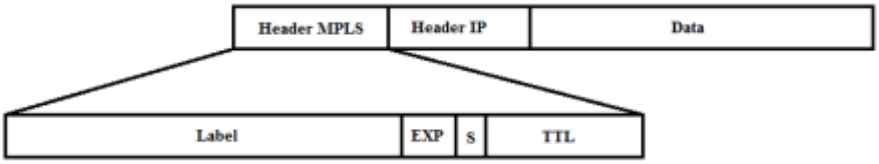
Label yang diterapkan pada setiap paket data memiliki peran penting dalam mengatur aliran paket antara dua titik akhir. Node-node dalam jaringan MPLS ini dikenal sebagai *Label Switched Router* (LSR), dan tugas utamanya adalah mengalihkan paket data saat melewati setiap *router*. LSR mampu menentukan jalur yang berbeda untuk setiap aliran paket, menciptakan jalur khusus yang disebut *Forwarding Equivalence Class* (FEC). Setiap jalur FEC ini memiliki karakteristik yang mempengaruhi kebutuhan kualitas layanannya (QoS).

Untuk paket yang memenuhi karakteristik tertentu, diperlukan penetapan jalur sebelumnya melalui jaringan, yang disebut sebagai *Label Switched Path* (LSP).



Gambar 2.9 Pengiriman Paket pada Jaringan MPLS.

Gambar 2.9 adalah cara kerja MPLS dalam mengirimkakn paket. Prinsip kerja MPLS adalah dengan menggabungkan kecepatan *switching* pada *layer* 2 dan kemampuan *routing* pada *layer* 3. MPLS meneruskan paket dengan menyelipkan label diantara *header layer* 2 dan *layer* 3. Label tersebut dihasilkan oleh *Label Switching Router* (LSR) yang berisi informasi tujuan node selanjutnya. Paket yang diteruskan disebut dengan *Label Switching Path* (LSP). LSP dibentuk oleh *protocol* pensinyalan yang menentukan *forwarding* berdasarkan label pada paket. *Router* dalam mengambil keputusan untuk melewatkan paket melalui suatu jalur menggunakan *Label Forwarding Information Base* (LFIB).



Gambar 2.10 *Header* MPLS.

MPLS hanya melakukan enkapsulasi paket dengan menambahkan *header* pada paket tersebut. *Header* MPLS terdiri dari 32 bit data yang termasuk 20 bit label, 3 bit *experiment*, 1 bit *stack*, dan 8 bit TTL (*Time to Live*). Header MPLS dapat dilihat pada gambar 2.10. *Header* MPLS berperan sebagai perekat antara *header layer* 2 dan *layer* 3. Label adalah bagian dari *header* yang digunakan

untuk proses *forwarding* dan *traffic engineering*. Berikut adalah komponen dari *header* MPLS :

1. *Label*

   Merupakan *field* yang terdiri dari 20 bit yang merupakan nilai dari label tersebut. Nilai label tersebut contohnya alamat IP, besar data, jenis data dan lain-lain.

2. *Experiment* (EXP)

   Secara teknis *field* ini digunakan untuk keperluan eksperimen yaitu untuk menunjukkan antrian data yang masuk dan penjadwalan pengiriman paket. EXP terdiri dari 3 bit.

3. *Bottom of Stack* (STACK)

   Sebuah paket memungkinkan menggunakan lebih dari satu label. *Field* ini digunakan untuk mengetahui label *stack* yang paling bawah. Label yang paling bawah dalam *stack* memiliki nilai bit 1 sedangkan yang lain diberi nilai bit 0. Hal ini sangat diperlukan pada proses label *stacking*.

4. *Time-to-Live* (TTL)

   Field ini biasanya merupakan hasil salinan dari IP TTL *header* yang membantu dalam proses pendeteksian dan penghentian *looping* dari paket MPLS [17].

2.5 *ROUTING PROTOCOL*

Proses pengalihan atau pengiriman data yang terjadi pada *router* jaringan yang disebut sebagai *routing*. *Router* akan memilih rute yang tepat untuk mengirimkan data ke tujuan yang diinginkan. Dalam proses ini, *router* akan mengelola informasi tentang rute-rute data tersebut dan menyimpannya dalam bentuk tabel *routing*. Tabel *routing* adalah suatu tabel yang digunakan dalam jaringan komputer untuk mengarahkan paket data dari satu perangkat ke perangkat lainnya. Tabel ini berisi daftar informasi tentang jaringan tujuan dan cara terbaik untuk mencapai jaringan tersebut. Saat suatu perangkat menerima paket data, tabel *routing* digunakan untuk menentukan jalur yang optimal untuk mengirim paket tersebut ke tujuan. Tabel *routing* berisi informasi sebagai berikut :

1. *Network Destination* adalah alamat jaringan tujuan dari paket data.

2. *Next Hop* adalah alamat IP perangkat jaringan berikutnya yang akan meneruskan paket ke arah tujuan.
3. *Metric* adalah nilai yang menggambarkan kondisi atau kualitas rute. Semakin rendah nilai metrik, semakin baik kualitas rute. Metrik bisa berupa jumlah *hop*, kecepatan, beban jaringan, dll.
4. *Interface* adalah antarmuka fisik atau logis pada *router* yang harus digunakan untuk mengirimkan paket ke *next hop* [18].

Tabel *routing* secara periodik diperbarui dan dikelola oleh protokol *routing*. Terdapat dua macam *routing*, yaitu:

1. *Routing* statis adalah pendekatan dalam pengaturan jaringan di mana para administrator secara langsung mengelola dan mengubah tabel *routing* secara manual. Dalam *routing* statis, administrator memasukkan detail yang terkait dengan jaringan, seperti alamat tujuan, ke dalam tabel *routing*. Kemudian, menentukan *port* atau *interface* khusus pada *router* yang akan berperan sebagai jalur pengirim data menuju tujuan. *Routing* statis memberikan kendali penuh kepada administrator jaringan dalam menentukan rute paket data, namun juga memerlukan pembaruan manual setiap kali terjadi perubahan dalam topologi jaringan atau kebutuhan *routing*.
2. *Routing* dinamis adalah suatu pendekatan dalam manajemen jaringan di mana protokol khusus digunakan untuk melakukan penelusuran dan penemuan jaringan secara otomatis. Protokol ini bertanggung jawab untuk mengidentifikasi rute-rute yang tersedia dalam jaringan yang terhubung dan secara dinamis memperbaharui tabel *routing* dengan informasi tentang rute-rute tersebut. Tabel *routing*, yang berperan sebagai peta navigasi bagi perangkat jaringan seperti *router*, terus-menerus diperbarui oleh protokol *routing* dinamis dengan informasi terbaru tentang keadaan jaringan, ketersediaan jalur, dan metrik yang terkait dengan setiap rute. Dengan adanya *routing* dinamis, jaringan dapat merespons perubahan topologi secara otomatis tanpa campur tangan manusia. Ketika ada perubahan dalam jaringan, seperti tambahan atau pemutusan koneksi, protokol *routing* dinamis akan mendeteksi perubahan tersebut dan melakukan pembaruan tabel *routing* sesuai dengan kondisi terbaru. Hal ini menghasilkan efisiensi

dalam pengiriman paket data, karena jaringan akan selalu menggunakan rute terbaik dan paling efisien berdasarkan informasi terkini yang diberikan oleh protokol *routing* dinamis [19].

Jaringan komputer umumnya menggunakan algoritma *routing* dinamis yang lebih kompleks. Terdapat dua algoritma dinamis, yaitu :

1. *Distance Vector*

Algoritma *routing distance vector* meminta setiap *router* dalam jaringan untuk memelihara tabel atau vektor yang berisi informasi tentang jarak terbaik ke setiap tujuan dan tautan yang harus digunakan. Informasi ini diperbarui melalui pertukaran data dengan *router* tetangga. Setiap *router* memiliki pengetahuan tentang tautan terbaik ke semua tujuan. Algoritma ini juga dikenal sebagai algoritma *routing Bellman-Ford* terdistribusi, yang awalnya digunakan dalam jaringan ARPANET dan dalam internet dengan nama RIP. Dalam algoritma ini, setiap *router* memiliki tabel *routing* yang berisi entri untuk setiap *router* lain dalam jaringan. Setiap entri mencakup tautan keluar yang diinginkan dan perkiraan jarak ke tujuan, yang dapat diukur dalam *hop* atau metrik lain. *Router* diasumsikan mengetahui jarak ke tetangga terdekatnya, baik dalam *hop* atau *delay* propagasi, yang dapat diukur dengan paket *echo* khusus.

2. *Link State*

Routing distance vector digunakan dalam ARPANET sampai tahun 1979, saat digantikan oleh *routing link state*. Algoritma ini digantikan karena masalah konvergensi yang lambat setelah perubahan topologi jaringan (*count-to-infinity*). Algoritma baru yang dikenal sebagai *routing link state*, termasuk varian seperti IS-IS dan OSPF, menjadi yang paling umum digunakan dalam jaringan besar [14].

Routing protocol adalah perangkat lunak yang bertugas untuk pertukaran informasi *routing* dan pembentukan tabel *routing*. Dalam tabel *routing* terdapat alamat tujuan paket data. Protokol *routing* terbagi menjadi dua jenis, yaitu IGP (*Interior Gateway Protocol*) dan EGP (*Exterior Gateway Protocol*). Protokol IGP digunakan ketika *router* dalam satu sistem atau otonomi secara bebas menukar informasi *routing* di dalam satu AS (*Autonomous System*). Contoh protokol IGP termasuk RIP, OSPF, EIGRP, dan IS-IS. Sedangkan protokol EGP digunakan

untuk menukar informasi antara AS yang berbeda. Contoh protokol EGP adalah BGP [20].

2.5.1 RIPng (*Routing Information Protocol New Generation*)

Routing Information *Protocol* (RIP) adalah sebuah protokol generasi *routing* yang pertama untuk IPv4 awalnya ditetapkan dalam RFC 1058. Sangat mudah untuk mengkonfigurasi, menjadikannya pilihan yang baik untuk jaringan kecil. Pada tahun 1993, RIPv1 telah diperbarui untuk protokol *routing* tanpa kelas (*Classless*) yang dikenal sebagai versi RIP 2 (RIPv2). *Update* RIP diringkas menjadi segmen UDP, dengan kedua sumber dan tujuan nomor *port* diatur ke *port* UDP 520. Pada tahun 1997, IPv6 versi diaktifkan dari RIP dirilis. RIPng didasarkan pada RIPv2. RIPng masih memiliki keterbatasan 15 *hop* dan jarak administrasi adalah 120 [21].

Kelas *Interior Gateway Protocol*, sering dikenal sebagai RIP dan dikembangkan oleh IETF, adalah salah satu contoh protokol *routing distance vector*. Protokol *routing* ini memilih jalur *routing* optimal menggunakan algoritma *Bellman-Ford*. RIP digunakan pada jaringan kecil karena mudah diatur dan dikonfigurasi. Iterasi terbaru dari RIP, RIPng, melakukan transportasi data saat mengirim atau menerima datagram melalui protokol UDP pada *port* 521. RIPng adalah salah satu protokol *routing distance vector* yang memilih rute ke tujuan dengan menghitung *hop*. Pada dasarnya cara kerja dari RIPng sama dengan RIP, karena RIPng dibuat berdasarkan *routing* protokol RIP yang digunakan pada IPv4. Namun salah satu perbedaan yang mendasar ialah dukungan pada pengalamatan IPv6 [20].

RIPng (*Routing Information Protocol next generation*) adalah versi dari protokol *routing* RIP yang dirancang khusus untuk penggunaan dengan protokol IPv6. RIPng memungkinkan pertukaran informasi rute IPv6 di seluruh jaringan. IPv6 adalah versi protokol Internet yang menggantikan IPv4 dan menggunakan alamat IPv6 yang lebih panjang. Karena perbedaan struktur alamat dan beberapa fitur protokol antara IPv4 dan IPv6, RIPng dikembangkan untuk memberikan dukungan khusus untuk alamat IPv6.

RIPng memiliki beberapa perbedaan dibandingkan dengan RIP (RIPv1 dan RIPv2) yang dirancang untuk IPv4. Beberapa perbedaan utama RIPng adalah sebagai berikut :

1. Penggunaan alamat IPv6

   RIPng dirancang untuk bekerja dengan alamat IPv6 yang terdiri dari 128 bit, sedangkan RIP klasik (RIPv1 dan RIPv2) digunakan untuk alamat IPv4 yang terdiri dari 32 bit.

2. *Metric*

   RIPng menggunakan metrik *hop count* seperti RIP klasik. Namun, dalam RIPng, metrik *hop count* dihitung berdasarkan jumlah *router* antara sumber dan tujuan, menggunakan nilai maksimum 16 *hop*.

3. Pengumuman otomatis

   RIPng menggunakan pengumuman otomatis untuk mengirimkan informasi rute IPv6 ke tetangga jaringan. Ini berarti setiap *router* RIPng akan secara otomatis mengumumkan rute-rute yang diketahuinya ke tetangganya.

4. Penanganan *subnetting* dan VLSM

   RIPng mendukung *subnetting* dan VLSM (*Variable Length Subnet Masking*) untuk IPv6. Ini berarti RIPng dapat mengenali dan mengirimkan informasi tentang *subnet-subnet* yang lebih kecil di dalam jaringan IPv6 yang lebih besar [22].

RIPng menggunakan *timer*, prosedur, dan tipe *message* yang sama dengan RIPv2. Misalnya, RIPv2 menggunakan *update timer* 30 detik yang telah ditambahi sedikit untuk mencegah sinkronisasi, periode *timeout* 180-detik, dan *timer* untuk *garbage- collection* 120 detik, dan *holddown timer* 180 detik. RIPng juga menggunakan *metric hop-count*, dengan 16 menunjukkan nilai *unreachable*. Dan juga menggunakan *Request* dan *Response messages* dengan cara yang sama seperti RIPv2. Serta pesan *Request* dan *Response* dikirim secara *multicast* dengan sedikit pengecualian untuk *unicast* yang digunakan RIPv1 dan v2. *Address multicast* IPv6 yang digunakan RIPng adalah FF02::9.

RIPng menentukan *address next-hop* dengan cara yang sama seperti RIPv2. Dengan kata lain, *address next-hop non-zero* yang valid menentukan *router next-hop* selain dari pengirim dari *message Response* dan *address next-hop*

0:0:0:0:0:0:0:0 menentukan pengirim dari *message Response* itu sendiri sebagai *address next-hop*. Bedanya, RIPng menentukan bahwa *address next-hop* pada entri route spesial kemudian mengelompokkan semua entri route yang menggunakan *address next-hop* setelahnya [23].

Prinsip kerja *routing* protokol RIPng melibatkan pertukaran informasi rute IPv6 antara *router* dalam jaringan. Berikut adalah prinsip kerja utama dari RIPng:

1. Pertukaran *Hello Messages*

   Ketika dua *router* RIPng pertama kali terhubung, kedua *router* tersebut bertukar pesan "Hello" untuk membangun koneksi dan mengidentifikasi diri.
2. Pengumuman rute: Setiap *router* RIPng secara periodik mengirim pengumuman rute ke semua tetangganya dalam jaringan. Pengumuman ini berisi daftar rute yang diketahui oleh *router* tersebut.
3. Penerimaan pengumuman: *Router* lain yang menerima pengumuman ini menyimpan informasi rute dalam tabel rute lokal mereka.
4. Perhitungan metrik: RIPng menggunakan metrik *hop count* (jumlah hop) sebagai cara untuk menentukan jarak relatif ke tujuan tertentu.
5. Pembaruan periodik: *Router* akan terus mengirimkan pembaruan periodik tentang tabel rute mereka dalam interval waktu tertentu. Pembaruan ini memastikan bahwa semua *router* memiliki informasi rute yang diperbarui dan akurat.
6. Konvergensi: Konvergensi adalah proses di mana semua *router* dalam jaringan mencapai pemahaman yang konsisten tentang topologi jaringan. Dalam RIPng, konvergensi terjadi ketika tabel rute semua *router* telah diperbarui dengan informasi rute yang paling baru dan konsisten.
7. Penyaringan rute: RIPng tidak memperhitungkan parameter kualitas jaringan seperti *bandwidth* atau *delay*.
8. Pemilihan rute terbaik: Dalam RIPng, rute dengan *hop count* terendah dianggap sebagai rute terbaik. Meskipun sederhana, ini tidak selalu menghasilkan jalur yang paling efisien dalam jaringan yang lebih besar dan kompleks.

Dengan prinsip-prinsip ini, RIPng memungkinkan *router* dalam jaringan untuk berkomunikasi dan bertukar informasi rute, sehingga data dapat diarahkan dengan benar melalui jalur terpendek menuju tujuan yang diinginkan.

2.5.2. BGP (*Border Gateway Protocol*)

Border Gateway System (BGP) dibuat terutama untuk memilih *rute* di seluruh domain (AS). Fungsi utama BGP adalah untuk mengkomunikasikan data tentang jangkauan jaringan antara *router* BGP. Perbedaan utama antara BGP dan protokol *routing* lainnya adalah bahwa BGP dirancang untuk merutekan antar AS daripada dalam satu AS. Jumlah perjalanan AS melalui saluran transmisi data adalah bagian dari informasi yang disampaikan. Dengan pengetahuan ini, dimungkinkan untuk membuat jaringan jalur AS yang saling berhubungan yang akan mencegah *loop* perutean.

BGP juga digunakan untuk menegakkan kebijakan perutean. Dengan membuat sesi berdasarkan koneksi TCP antara satu *router* BGP dan *router* BGP lainnya, informasi perutean dipertukarkan. Ada dua jenis koneksi BGP: eBGP (BGP eksternal), yang digunakan untuk mengumumkan awalan yang dapat dikunjungi oleh *router* BGP terdekat dari AS lainnya, dan iBGP (BGP internal), yang berfungsi untuk mengirimkan rute antara *router* internal dalam satu AS. Atribut jalur AS tidak berubah dalam transmisi informasi antara iBGP dan eBGP, yang merupakan perbedaan satu sama lain.

Autonomous system merupakan suatu set *routing* dalam domain yang dikelola oleh satu otoritas sehingga pengaruhnya dapat langsung diketahui oleh *router* maupun *peer – router*. Dengan adanya informasi ini, dapat dibentuk grafik dari AS *path* yang saling terkoneksi sehingga dapat menghindari terjadinya *routing loop* [2].

BGP adalah protokol *routing* eksternal yang digunakan untuk berkomunikasi antara *Autonomous System* yang berbeda (AS), dan juga disebut sebagai *routing* di seluruh domain ketika terjadi dalam satu *Autonomous System* (AS). BGP adalah protokol *path vector routing*, yang dikenal juga sebagai protokol *routing distance vector*. *Path Vector Protocol* adalah jenis protokol *routing* yang digunakan untuk pertukaran informasi *routing* di jaringan.

Karakteristik utama dari protokol jenis ini adalah bahwa setiap pesan yang dikirim oleh protokol tersebut menyertakan informasi tentang jalur atau *vector* yang diambil oleh paket untuk mencapai tujuan tertentu.

Dalam *path vector protocol*, setiap *router* akan menyimpan informasi tentang jalur yang dilalui oleh paket menuju tujuan. Informasi ini disebut sebagai "*path vector*." Setiap vektor jalur terdiri dari serangkaian AS yang dilewati oleh paket, dan setiap AS dalam urutan itu biasanya dicatat bersama dengan nomor AS-nya. Dengan menyimpan informasi ini, protokol dapat mendeteksi potensi *loop routing* yang tidak diinginkan.

Protokol *routing* yang menggunakan pendekatan *path vector* umumnya lebih cocok untuk lingkungan jaringan yang kompleks, di mana ada banyak jalur yang berpotensi menuju tujuan. Penting untuk diingat bahwa karakteristik *path vector*, memiliki manfaat dalam menghindari *loop routing*, juga dapat menghadirkan tantangan tertentu dalam konvergensi jaringan dan pengendalian jalur yang rumit.

Kelebihan *protocol routing* BGP adalah sebagai berikut :

1. Skalabilitas: BGP dirancang untuk mengatasi jaringan yang sangat besar dan kompleks.
2. Fleksibilitas: BGP memberikan kontrol yang sangat baik terhadap jalur yang diambil oleh paket. Administrator dapat menerapkan kebijakan *routing* yang sangat detail dan kompleks berdasarkan berbagai faktor seperti biaya, kualitas jalur, atau preferensi jalur tertentu.
3. Stabilitas: BGP dirancang untuk menjaga stabilitas jaringan. Perubahan rute biasanya diumumkan secara hati-hati, dan protokol ini mencoba untuk menghindari perubahan yang tiba-tiba atau *looping*.
4. Dukungan *Multi-Homed Networks*: BGP sangat cocok untuk jaringan yang terhubung ke beberapa penyedia layanan internet (*multi-homed*). Ini memungkinkan jaringan untuk mengoptimalkan lalu lintas dan memastikan konektivitas yang andal.
5. IPv6 *support*: BGP mendukung pengalamatan IPv6, memungkinkan pertukaran informasi rute untuk jaringan yang menggunakan versi protokol ini [24].

Selain memiliki kelebihan, *routing* BGP memiliki tantangan dan resiko. Kekurangan *protocol routing* BGP antara lain :

1. Konvergensi yang lambat: BGP memiliki waktu konvergensi yang lebih lambat dibandingkan dengan beberapa protokol *routing* lainnya. Ini terutama terjadi saat terjadi perubahan jaringan yang signifikan.
2. Konfigurasi rumit: Konfigurasi BGP dapat menjadi kompleks dan memerlukan pemahaman yang mendalam tentang topologi jaringan dan kebijakan *routing*.
3. Potensi serangan: Karena fleksibilitasnya, BGP rentan terhadap serangan seperti BGP hacking, di mana rute palsu diumumkan dan lalu lintas dialihkan secara tidak sah.
4. Tabel rute besar: Meskipun BGP dapat mengelola banyak rute, semakin besar tabel rute BGP dapat memakan sumber daya memori pada *router*.

Cara kerja *routing* BGP (*Border Gateway Protocol*) melibatkan pertukaran informasi *routing* antara *router* di dalam AS yang berbeda. BGP digunakan untuk memutuskan jalur terbaik yang harus diambil oleh paket data untuk mencapai tujuannya. Berikut adalah langkah-langkah *routing* BGP bekerja:

1. Pertukaran *Hello Messages*

   Ketika dua *router* BGP pertama kali terhubung, kedua *router* tersebut bertukar pesan "Hello" untuk membangun koneksi dan mengidentifikasi diri.
2. Pertukaran informasi pengumuman rute

   Setelah koneksi terbentuk, *router* BGP akan mulai menukar informasi rute. Mereka mengumumkan rute-rute yang mereka miliki kepada *router* tetangga. Informasi ini berisi tentang *subnet*, nomor AS pengirim, atribut *path*, dan lain-lain.
3. Pemilihan rute terbaik

   Setiap *router* BGP akan menerima informasi rute dari tetangganya dan memutuskan rute mana yang akan diambil berdasarkan kebijakan *routing* yang dikonfigurasi.
4. Pengumuman rute terpilih

   Router BGP kemudian mengumumkan rute-rute yang mereka pilih kepada tetangga-tetangganya.

5. *Update* periodic

   BGP terus memantau perubahan dalam topologi jaringan dan rute yang tersedia. Jika ada perubahan, seperti rute menjadi tidak tersedia atau ada jalur baru yang muncul, *router* akan mengirimkan pembaruan kepada tetangga mereka.

6. Konvergensi

   Konvergensi adalah proses di mana *router* BGP dalam AS mencapai keadaan di mana mereka semua memiliki informasi rute yang konsisten.

7. *Loop Prevention*

   BGP menggunakan *path vector* untuk mencegah *loop routing*. Informasi jalur yang disertakan dalam pesan BGP membantu mengidentifikasi jalur yang telah dilewati dan menghindari rute yang membentuk *loop*.

8. Pemeliharaan

   Router BGP terus memelihara koneksi dengan tetangganya dan memastikan informasi rute tetap diperbarui dan konsisten.

Melalui proses ini, BGP memungkinkan *router* untuk mengambil keputusan tentang bagaimana mengarahkan lalu lintas di jaringan yang kompleks dan terdistribusi. Keputusan ini diambil berdasarkan kebijakan dan preferensi yang dikonfigurasi oleh administrator jaringan. Untuk membentuk dan mempertahankan sebuah sesi BGP dengan *router* tetangganya, BGP mempunyai mekanismenya sendiri. Pembentukan sesi BGP ini mengirimkan paket-paket pesan yang terdiri dari empat macam. Paket-paket tersebut adalah :

1. *Open Message*

   Sesuai dengan namanya, paket pesan jenis ini merupakan paket pembuka sebuah sesi BGP. Paket inilah yang pertama dikirimkan ke *router* tetangga untuk membangun sebuah sesi komunikasi. Paket ini berisikan informasi mengenai BGP *version number*, AS *number*, *hold time*, dan *router* ID.

2. *Keepalive Message*

   Paket *Keepalive message* bertugas untuk menjaga hubungan yang telah terbentuk antarkedua *router* BGP. Paket jenis ini dikirimkan secara periodik

oleh kedua buah *router* yang bertetangga. Paket ini berukuran 19 byte dan tidak berisikan data sama sekali.

3. *Notification Message*

   Paket pesan ini adalah paket yang bertugas menginformasikan error yang terjadi terhadap sebuah sesi BGP. Paket ini berisikan *field-field* yang berisi jenis *error* apa yang telah terjadi, sehingga sangat memudahkan penggunanya untuk melakukan *troubleshooting*.

4. *Update Message*

   Paket update merupakan paket pesan utama yang akan membawa informasi rute-rute yang ada. Paket ini berisikan semua informasi rute BGP yang ada dalam jaringan tersebut. Ada tiga komponen utama dalam paket pesan ini, yaitu *Network-Layer Reachability Information* (NLRI), *path attribut*, dan *withdrawn routes*.

2.6 QOS (*QUALITY OF SERVICE*)

Memastikan layanan yang sesuai adalah salah satu persyaratan paling penting bagi jaringan telekomunikasi modern. *International Telecommunication Union* atau ITU mendefinisikan *Quality of Service* merupakan sekelompok layanan telekomunikasi yang terkait dengan kemampuannya untuk memenuhi kebutuhan yang telah ditetapkan dan diantisipasi oleh pengguna layanan. Sangat penting untuk mengembangkan metode untuk memasukkan umpan balik pelanggan ketika menilai Kualitas Layanan [25]. Kemampuan jaringan untuk memberikan layanan berkualitas tinggi dengan menyediakan *bandwidth* yang cukup dan mengatasi masalah *jitter* dan latensi disebut sebagai QoS (*Quality of Service*). Pengukuran QoS menjelaskan seberapa cepat dan andal berbagai jenis data ditransmisikan selama komunikasi. Faktor QoS mencakup hal-hal seperti *throughput*, *packet loss*, *latency*, *jitter* (variasi dalam kedatangan paket), dan MOS (*Mean Opinion Score*) [26].

2.6.1 *Packet loss*

Rasio paket yang ditransmisikan dalam jaringan yang tidak berhasil diterima dikenal sebagai *packet loss* [25]. Data yang diterima sering disimpan

dalam *buffer* pada perangkat jaringan. *Buffer* akan terisi jika kemacetan berlangsung lama, sehingga tidak mungkin untuk menerima data baru. % adalah pengukuran yang digunakan untuk menghitung kehilangan paket. Versi sederhana rumus perhitungan kehilangan paket berikut tersedia [27] :

$$Packet\ Loss\ = \frac{Paket\ data\ dikirim - Paket\ data\ diterima}{(Paket\ data\ yang\ dikirim)}\ x\ 100$$

Pengklasifikasian *packet loss* dari suatu jaringan dapat mengacu pada standarisasi TIPHON-05001 seperti yang tercantum dalam Tabel 2.2:

Tabel 2.2 Standarisasi *packet loss* berdasarkan TIPHON

Category	*Packet loss*
Sangat Bagus	0 %
Bagus	3 %
Sedang	15 %
Buruk	25 %

2.6.2 *Throughput*

Throughput merupakan jumlah *byte frame* yang diterima dalam jaringan dalam periode waktu tertentu [25]. Satuan yang digunakan dalam perhitungan *throughput* adalah bit per detik (bps) [20]. Periode *throughput* waktu tertentu dihitung sebagai jumlah total paket yang berhasil dikirim dan diterima di tujuan dibagi dengan durasi periode tersebut. Persamaan berikut dapat digunakan untuk menghitung nilai *throughput* [18].

$$Throughput\ = \frac{Paket\ data\ dikirim\ (bits)}{Jumlah\ Waktu\ pengiriman\ data\ (s)}\ (2)$$

2.6.3 *Delay*

Jumlah waktu yang dibutuhkan untuk data atau paket untuk pergi dari satu komputer ke komputer lain dikenal sebagai *delay*. Dalam jaringan komputer, menunggu lama atau pilihan rute yang berbeda untuk menghindari kemacetan *routing* dapat membuat penundaan dalam transmisi paket. Waktu pemrosesan yang lama, media fisik, lalu lintas, dan jarak semuanya dapat berkontribusi pada

penundaan. Seseorang dapat memperoleh penundaan transmisi dengan membagi panjang paket (dalam bit) dengan *bandwidth* tautan (dalam bit per detik). Rumus digunakan, seperti yang ditunjukkan pada persamaan di bawah ini, untuk mendapatkan penundaan rata-rata [18].

$$Delay = \frac{Total\ Delay\ atau\ waktu\ pengiriman\ data\ (s)}{Jumlah\ paket}$$

Tabel 2.3 Standarisasi *Delay* berdasarkan TIPHON

Category	*Delay*
Sangat Bagus	<150 ms
Bagus	150 s/d 300 ms
Sedang	300 s/d 450 ms
Buruk	>450 ms

2.7 GNS3

GNS3 (*Graphical Network Simulator*-3) GNS3 adalah emulator perangkat lunak jaringan yang digunakan untuk mensimulasikan jaringan kompleks dengan memungkinkan kombinasi perangkat virtual dan nyata [28].

GNS3 dapat digunakan untuk menguji berbagai jenis peralatan, bukan hanya peralatan dari satu produsen tertentu. GNS3 juga bersifat *multiplatform* dan dapat di*instal* pada sistem operasi *Windows*, *Linux*, dan *Mac*. GNS3 menggunakan kerangka kerja dari program *open-source*, di mana setiap program dapat mensimulasikan *platform* tertentu untuk menjalankan perangkat lunak yang sesuai dengan penyedia layanan. GNS3 digunakan untuk menguji konfigurasi yang akan diterapkan di masa depan pada peralatan nyata [29].

2.8 *WIRESHARK*

Wireshark adalah perangkat lunak sumber terbuka untuk yang digunakan untuk menganalisis dan menyelesaikan masalah dalam sebuah jaringan [28]. *Wireshark* juga digunakan untuk menganalisis perilaku jaringan setelah adanya celah keamanan yang dapat dieksploitasi (*backdoor exploitable*). Selain itu, *Wireshark* dapat digunakan untuk menghitung waktu *respons server web* dengan mengukur aliran data antara *server web* dan *client* [30].

Sering disebut sebagai analisis paket jaringan, *Wireshark* memiliki kemampuan untuk menangkap data atau paket jaringan dan menampilkan informasi tersebut secara rinci. Aplikasi ini umumnya digunakan oleh administrator jaringan untuk mengatasi masalah jaringan, mengamankan jaringan, dan mengembangkan jaringan dengan menerapkan beberapa protokol [31].

2.9 *VMWARE WORKSTATION PRO*

Vmware Workstation Pro adalah salah satu *software* virtualisasi yang paling populer digunakan oleh perusahaan atau pekerja IT. *software* ini memungkinkan para pekerja IT untuk membangun dan menguji mesin virtual dan perangkat jaringan yang mirip dunia nyata dari PC/laptop. *VMware Workstation* juga merupakan serangkaian produk desktop yang memungkinkan pengguna menjalankan mesin *virtual* [32].