BAB 2

LANDASAN TEORI

Pada bab 2 ini membahas mengenai teori jaringan komputer dan teori khusus tentang MPLS dan *Traffic Engineering*.

Jaringan komputer merupakan sekumpulan komputer, serta perangkat-perangkat lain pendukung komputer yang saling terhubung dalam suatu kesatuan. Peralatan jaringan yang digunakan ialah *switch* dan *router*. Serta kabel yang digunakan ialah *twisted pair*, kabel merupakan salah satu bagian yang terpenting dalam media koneksi antara komputer dengan komputer lainnya, setiap jenis kabel mempunyai kemampuan dan spesifikasi yang berbeda.

Topologi jaringan menjelaskan hubungan geometris antara unsur-unsur dasar penyusun jaringan yaitu *node, link*, dan *station*. Topologi yang digunakan *mesh*, topologi jaringan ini menerapkan hubungan antar sentral secara penuh. Setiap host memiliki hubungan langsung dengan semua host lainnya dalam jaringan. Topologi ini juga merefleksikan internet yang memiliki banyak jalur ke satu titik.

Berdasarkan jangkauannya menggunakan *Wide Area Network* (WAN)` merupakan jaringan komputer yang mencakup area yang besar sebagai contoh yaitu jaringan komputer antar wilayah, kota atau bahkan negara, atau dapat didefinisikan juga sebagai jaringan komputer yang membutuhkan *router* dan saluran komunikasi publik.

Multiprotocol Label Switching (disingkat menjadi **MPLS**) adalah teknologi penyampaian paket pada jaringan *backbone* berkecepatan tinggi. Sedangkan *traffic engineering* memanipulasi trafik agar sesuai dengan jaringan.

2.1 Teori Umum

2.1.1 Pengertian Jaringan

Menurut Sopandi(2010, p2), Jaringan komputer merupakan sekumpulan komputer, serta perangkat-perangkat lain pendukung komputer yang saling terhubung dalam suatu kesatuan. Gabungan teknologi ini melahirkan pengolahan data yang dapat didistibusikan mencakup pemakaian bersama, sehingga penggunaan komputer yang sebelumnya hanya berdiri sendiri, kini telah diganti dengan sekumpulan komputer yang terpisah-pisah akan tetapi saling berhubungan dalam melaksanakan tugasnya, sistem seperti inilah yang disebut jaringan komputer (*computer network*).

Internetwork merupakan kumpulan jaringan individu yang terhubung dengan peralatan jaringan dan yang berfungsi sebagai sebuah jaringan yang besar. Internet umum merupakan contoh yang paling umum, dimana sebuah jaringan menghubungkan jutaan komputer.

Tujuan dari jaringan komputer antara lain membagi sumber daya (berbagi pemakaian printer, CPU, memori, *harddisk*), berkomunikasi (seperti e-*mail*, *instant messaging*, *chatting*), dan akses informasi (contohnya *web browsing*).

2.1.2 Peralatan Jaringan

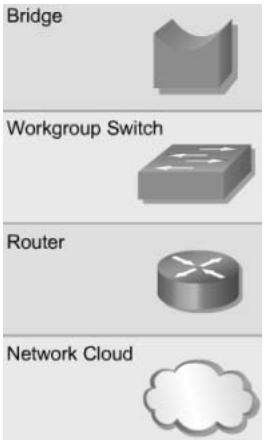
- ***Switch*(multi-port bridge)**

Menurut Sopandi(2010, p18), *Switch* bekerja pada *Layer 2* model OSI dan sebuah alat jaringan yang melakukan *bridging* transparan

(penghubung segementasi banyak jaringan dengan *forwarding* berdasarkan alamat MAC). *Switch* harus meneruskan *frame broadcast*. *Switch* membagi *collision domain* tetapi tidak membagi *broadcast domain*. *Switch* ada juga yang bekerja pada *Layer* 3. *Switch* ini meneruskan paket berdasarkan informasi *Layer* 3 dan biasanya digunakan untuk jaringan LAN.

- ***Router***

  Router berfungsi sebagai penghubung antar dua atau lebih jaringan untuk meneruskan data dari satu jaringan ke jaringan lainnya. Sehingga di setiap port yang dimiliki sebuah *router* harus memiliki alamat IP yang berbeda jaringan. *Router* bekerja pada *Layer* ketiga model OSI. *Router* membagi *collision domain* dan *broadcast domain*.



Gambar 2.1 Simbol Peralatan Jaringan
(Sumber: http//cnap.binus.ac.id/)

- **Kabel dan konektor**

Menurut Sopandi(2010, p20), Kabel merupakan salah satu bagian yang terpenting dalam media koneksi antara komputer dengan komputer lainnya, setiap jenis kabel mempunyai kemampuan dan spesifikasi yang berbeda. Jenis kabel tersebut yaitu:

1. *Twisted Pair Ethernet*

   Kabel *Twisted Pair* ini terbagi menjadi dua jenis yaitu STP (*shielded twisted pair*) dan UTP (*unshielded twisted pair*). *Shielded* adalah jenis kabel yang memiliki selubung pembungkus. Untuk koneksinya kabel jenis ini menggunakan konektor RJ-11 atau RJ-45. Pada *twisted pair* (10 BaseT) *network*, komputer disusun membentuk suatu pola star. Setiap PC memiliki satu kabel *twisted pair* yang tersentral pada HUB.

Gambar 2.2 kabel twisted pair ethernet
(Sumber: http://syafran.wordpress.com)

2. *Coaxial Cable*

    1. *Thin coaxial cable* (Kabel *Coaxial* "Kurus")

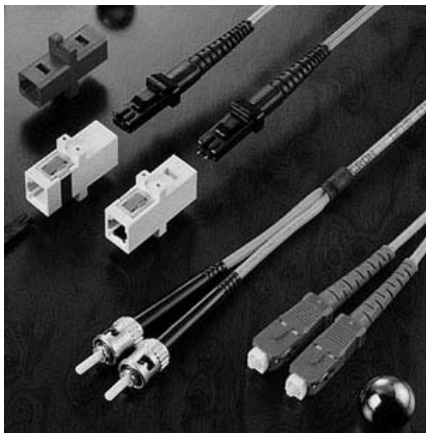
        Thin Ethernet atau *Thinnet* memiliki keunggulan dalam hal biaya yang relatif lebih murah dibandingkan dengan tipe pengabelan lain, serta pemasangan komponennya lebih mudah. Panjang kabel *thin coaxial*/RG-58 antara 0.5-185 m dan maksimum 30 komputer terhubung.

    2. *Thick coaxial cable* (Kabel *Coaxial* "gemuk")

        Pada *Thick Ethernet* digunakan *transceiver* untuk menghubungkan setiap komputer dengan sistem jaringan dan konektor yang digunakan adalah konektor tipe DIX. Panjang kabel *transceiver* maksimum 50 m, panjang kabel *Thick Ethernet* maksimum 500 m dengan maksimum 100 *transceiver* terhubung.

3. *Fiber Optic*

    Jaringan *Fiber Optic* mempunyai kecepatan transfer data lebih dari 100Mbps dan dari segi kehandalan tidak diragukan. Berbeda dengan media transmisi lainnya, pada serat optik, gelombang pembawanya bukan gelombang electromagnet atau listrik, akan tetapi sinar/cahaya laser, sehingga tidak ada intervensi.



Gambar 2.3 Kabel dan *connector* FO

2.1.3 Klasifikasi Jaringan Komputer

2.1.3.1 Topologi Jaringan

Menurut Sopandi(2010, p27), Topologi jaringan menjelaskan hubungan geometris antara unsur-unsur dasar penyusun jaringan, yaitu *node*, *link*, dan *station*. *Physical topology* yang umumnya digunakan, antara lain:

- *Bus*

  Topologi jaringan bus merupakan beberapa simpul/*node* dihubungkan dengan jalur data (bus). Topologi Bus menyediakan 1 saluran untuk komunikasi semua perangkat sehingga setiap perangkat harus bergantian menggunakan saluran tersebut.

- *Ring*

  Topologi jaringan dimana setiap titik terkoneksi ke dua titik lainnya, sehingga membentuk suatu *loop* tertutup.

- *Star*

  Merupakan bentuk topologi jaringan yang berupa konvergensi dari *node* tengah ke setiap *node* atau pengguna. Topologi jaringan bintang termasuk topologi jaringan dengan biaya menengah.

- *Extended Star*

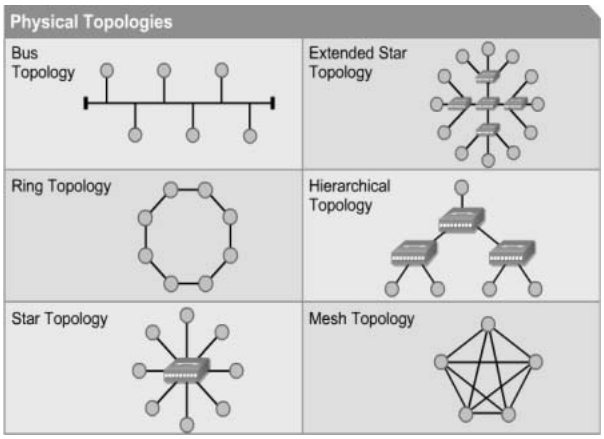
  Menggabungkan beberapa topologi *star* menjadi satu. Hub atau *switch* yang dipakai untuk menghubungkan beberapa komputer pada satu jaringan dengan menggunakan topologi star dihubungkan lagi ke hub atau *switch* utama.

- *Mesh*

  Topologi jaringan ini menerapkan hubungan antar sentral secara penuh. Setiap *host* memiliki hubungan langsung dengan semua host lainnya dalam jaringan. Topologi ini juga merefleksikan internet yang memiliki banyak jalur ke satu titik.

- *Hierarchical*

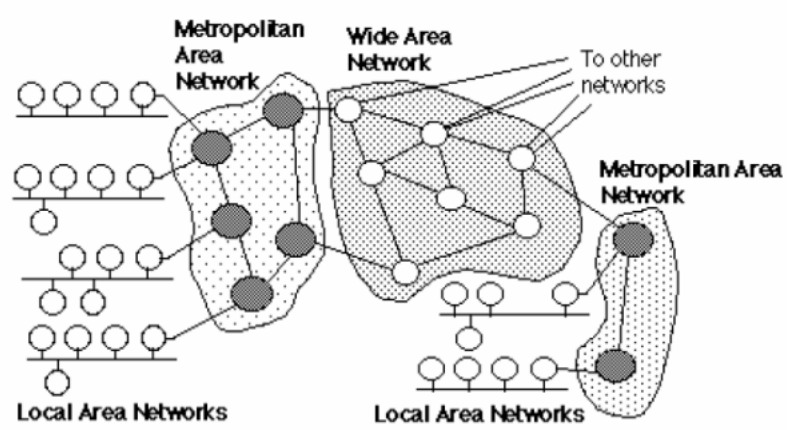
  Dibuat mirip dengan topologi *extended star* tetapi pada sistem jaringan yang dihubungkan dapat mengontrol arus data.



Gambar 2.4 *Physical Topologies*
(Sumber: http//cnap.binus.ac.id/)

2.1.3.2 Berdasarkan Jangkauan Geografis

Berdasarkan dari jangkauan geografis, jaringan komputer terbagi menjadi tiga ukuran, yaitu *Local Area Network (LAN)*, *Metropolitan Area Network (MAN)*, dan *Wide Area Network (WAN)*.



Gambar 2.5 Jaringan berdasarkan Geografisnya
(Sumber: http//cnap.binus.ac.id/)

2.1.3.2.1 Local Area Networks

Menurut Tanenbaum(2003, p16), Local Area Networks lebih dikenal dengan sebutan LAN, yaitu

jaringan milik pribadi dalam suatu bangunan atau kampus dan hanya mampu menjangkau sampai beberapa kilometer.

2.1.3.2.2 Metropolitan Area Networks

Menurut Tanenbaum(2003, p18), Metropolitan Area Networks lebih dikenal dengan sebutan MAN, merupakan jaringan yang jangkauannya mencapai ukuran sebuah kota, contohnya seperti jaringan televisi kabel.

2.1.3.2.3 Wide Area Networks

Menurut Tanenbaum(2003, p19-20), Wide Area Networks lebih dikenal dengan sebutan WAN, meliputi area yang sangat luas, biasanya mencapai ukuran negara atau benua.

2.1.4 Konsep *Networking* Model

Menurut Sopandi(2010, p53), Pada saat network baru muncul, kebanyakan komputer hanya bisa berkomunikasi dengan komputer yang dibuat oleh perusahaan yang sama. Untuk itu *International Organization for Standarization* membuat model referensi *Open System Interconnection* (OSI) sebagai solusi untuk mengatasi masalah kompabilitas ini.

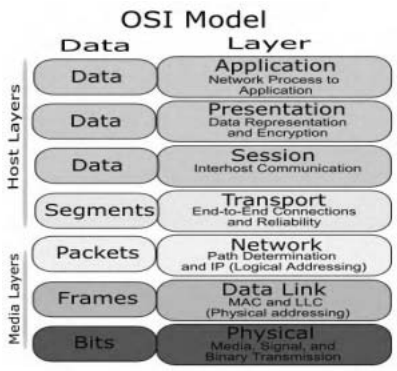
2.1.4.1 Model OSI *Layer*

Model referensi jaringan terbuka OSI atau ***OSI Reference Model for open networking*** adalah sebuah model arsitektural jaringan yang dikembangkan oleh badan International Organization for Standardization (ISO) di Eropa pada tahun 1977. OSI sendiri merupakan singkatan dari ***Open System Interconnection***. Model OSI terbagi dalam tujuh lapisan atau *Layer*, antara lain:

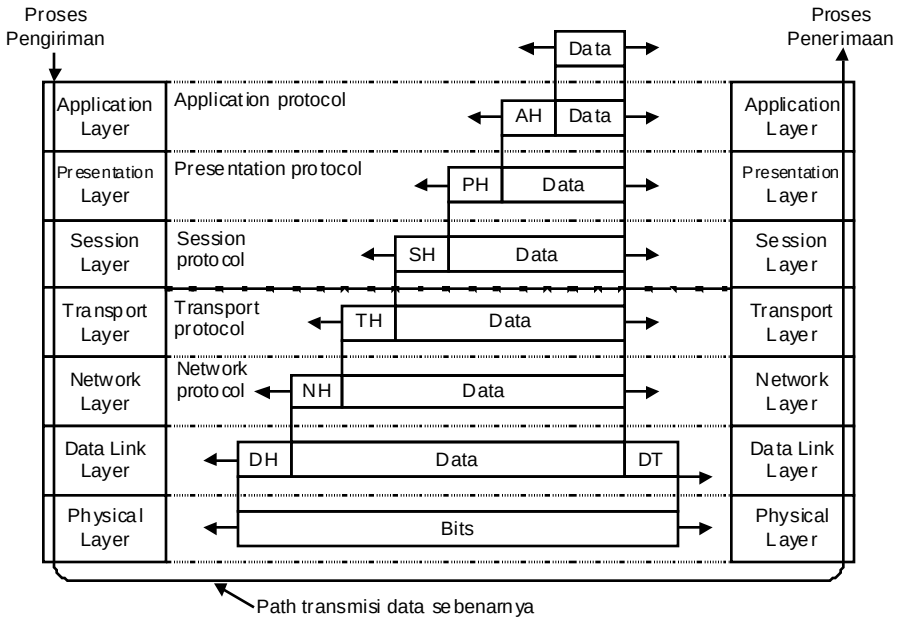
1. ***Physical Layer* (*Layer* 1)** Bertanggung jawab atas proses data menjadi bit dan mentransfernya melalui media, seperti kabel, dan menjaga koneksi fisik antar sistem. Selain itu, level ini juga mendefinisikan bagaimana Network Interface Card (NIC) dapat berinteraksi dengan media kabel atau radio.
2. ***Data Link Layer* (*Layer* 2)** Menyediakan link untuk data, memaketkannya menjadi frame yang berhubungan dengan "*hardware*" kemudian diangkut melalui media komunikasinya dengan kartu jaringan, mengatur komunikasi layer physical antara sistem koneksi dan penanganan *error*.
3. ***Network Layer* (*Layer* 3)** *Network* layer 3 berfungsi untuk mendefinisikan alamat-alamat IP, membuat *header* untuk paket-paket, dan kemudian melakukan *routing* melalui *internetworking* dengan menggunakan *router* dan *switch layer* 3.
4. ***Transport Layer* (*Layer* 4)** Berfungsi untuk memecah data ke dalam paket-paket data serta memberikan nomor urut ke paket-

paket tersebut sehingga dapat disusun kembali pada sisi tujuan setelah diterima. Bertanggung jawab membagi data menjadi segmen, menjaga koneksi logika "*end-to-end*" antar terminal, dan menyediakan *error handling* (penanganan error).

5. ***Session Layer* (*Layer* 5)** Session layer berfungsi untuk mendefinisikan bagaimana koneksi dapat dibuat, dipelihara atau dihancurkan. Selain itu, di level ini juga dilakukan resolusi nama.
6. ***Presentation Layer* (*Layer* 6)** *Presentation layer* berfungsi untuk mentranslasikan data yang hendak ditransmisikan oleh aplikasi ke dalam format yang dapat ditransmisikan melalui jaringan.
7. ***Application Layer* (*Layer* 7)** *Application layer* berfungsi sebagai *interface* dari aplikasi dengan fungsionalitas jaringan, mengatur bagaimana aplikasi dapat mengakses jaringan, dan kemudian membuat pesan-pesan kesalahan. Protokol yang berada dalam lapisan ini adalah HTTP, FTP, SMTP, dan NFS.



Gambar 2.6 Model OSI
(Sumber: http//cnap.binus.ac.id/)

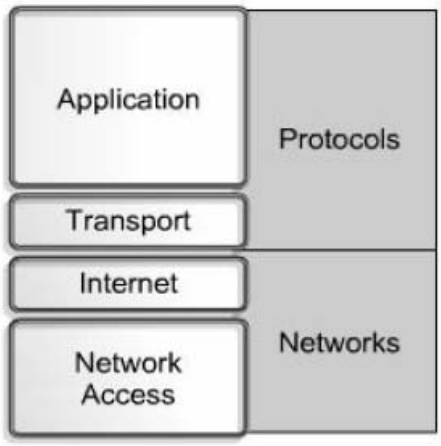


Gambar 2.7 Transmisi Data Pada Model OSI
(Sumber: http//cnap.binus.ac.id/)

2.1.4.2 Model TCP/IP *Layer*

Menurut Sopandi(2010, p60), *Transmission Control Protocol/Internet Protocol* (TCP/IP) merupakan standar industri protokol yang didesain untuk *Wide Area Network* (WAN). Model ini tidak sama dengan OSI Model meskipun keduanya mempunyai tujuan yang sama, yaitu sebagai fasilititator komunikasi antara dua pabrikan komputer dan model komputer, serta *operating system* yang berbeda. TCP/IP ini terdiri atas empat *layer*. Model Referensi *Transmission Control Protocol/Internet Protocol* (TCP/IP) diciptakan oleh Departemen Pertahanan Amerika (DARPA) karena mereka menginginkan jaringan yang dapat bertahan dalam kondisi apapun, sekalipun perang nuklir. *Department of Defense* (DOD) menginginkan *network* yang dapat mengrimkan paket setiap saat, dalam kondisi apapun, dari satu titik ke

titik lainnya. Masalah desain yang sangat sulit inilah yang menghasilkan Model TCP/IP, yang mana menjadi standar pertumbuhan internet. Model TCP/IP Memiliki 4 *layer*: *layer* Aplikasi, l*ayer Transport*, Internet *layer*, dan l*ayer Network Access*. Penting untuk diperhatikan bahwa beberapa *layer* pada Model TCP/IP memiliki nama yang sama dengan *layer* pada Model OSI.



Gambar 2.8 Model TCP/IP *Layer*
(Sumber: http//cnap.binus.ac.id/)

1. ***Application Layer*** Layer ini terdiri dari berbagai aplikasi yang dapat digunakan melalui jaringan seperti : HTTP, FTP, SMTP, Telnet, dan DNS. Bila dibandingkan dengan OSI Layer, layer ini merupakan gabungan dari *Application Layer* dengan *Presentation Layer* dan *Session Layer*. Pada l*ayer* ini akan terjadi data *compressed* dan data *uncompressed*.
2. ***Transport Layer*** Layer ini menyediakan sesi komunikasi antar dua komputer. *Layer* ini akan mendefinisikan layanan

transport yang digunakan, yaitu menggunakan *connection oriented* (TCP) atau *connectionless datagram oriented* (UDP).

- *Transmission Control Protocol* (TCP)

  Merupakan suatu layanan pengiriman berorientasi koneksi yang dapat diandalkan. Data TCP ditransmisikan dalam segmen-segmen dan suatu sesi harus ditetapkan sebelum *host* dapat mempertukarkan data. TCP memakai komunikasi byte-stream, yang berarti bahwa data diperlakukan sebagai suatu rangkaian byte. TCP mampu mencapai keterandalannya dengan menugaskan rangkaian angka ke setiap segmen yang ditransmisikan. Jika suatu segmen dibagi menjadi potongan-potongan yang lebih kecil, maka *host* penerima mengerti apakah semua potongan itu sudah diterima. Suatu pengakuan akan memverifikasi bahwa *host* lain sudah menerima data itu. Bagi setiap segmen yang dikirimkan, *host* penerima harus menghasilkan *acknowledgment* (ACK) dalam periode tertentu. Bila pengirim tidak menerima ACK, maka data tersebut ditransmisikan ulang. Kalau segmen yang diterima ternyata rusak, maka *host* penerima akan membuangnya. Karena dalam kasus ini ACK tidak dikirimkan, maka pengirim mentransmisikan ulang segmen itu.

- *User Datagram Protocol* (UDP)

  Menawarkan suatu layanan datagram tanpa koneksi yang menjamin entah pengiriman atau pengurutan paket-paket yang dikirimkan secara benar. *Checksum* data UDP bersifat opsional, yang menyediakan suatu cara untuk mempertukarkan data pada jaringan-jaringan yang sangat diandalkan tanpa perlu membutuhkan waktu pemrosesan atau sumber daya jaringan. UDP dipakai oleh aplikasi-aplikasi yang tidak memerlukan pengakuan tentang kuitansi data. Aplikasi tersebut secara khusus mentransmisikan sejumlah kecil data pada suatu waktu. Paket-paket yang disiarkan harus memakai UDP. Contoh layanan dan aplikasi yang memakai UDP adalah DNS, RIP, dan SNMP.

3. ***Internetwork Layer*** Melakukan enkapsulasi paket data menjadi internet datagram dan melakukan semua algoritma *routing*.
4. ***Networks Interface Layer*** adalah *level* yang paling bawah dari susunan TCP/IP. *Layer* ini adalah *device driver* yang memungkinkan *datagram* IP dikirim ke atau dari *phisycal network*.

2.1.5 Protocol TCP/IP

Transmission Control Protocol/Internet Protocol (TCP/IP) adalah standar komunikasi data yang digunakan oleh komunitas internet dalam proses tukar-menukar data dari satu komputer ke komputer lain di dalam jaringan internet. Protokol ini tidaklah dapat berdiri sendiri, karena memang protokol ini berupa kumpulan protokol (*protocol suite*). Protokol ini juga merupakan protokol yang paling banyak digunakan saat ini. Data tersebut diimplementasikan dalam bentuk *software* (perangkat lunak) di sistem operasi. Istilah yang diberikan kepada perangkat lunak ini adalah TCP/IP stack. Protokol ini juga bersifat *routable* yang berarti protokol ini cocok untuk menghubungkan sistem-sistem berbeda (seperti Microsoft Windows dan keluarga UNIX) untuk membentuk jaringan yang heterogen. TCP/IP merupakan kombinasi dari dua protokol terpisah. IP adalah protokol *Layer* 3 - suatu *service connectionless* yang menyediakan layanan pengantar data terbaik dalam jaringan. TCP adalah protokol *Layer* 4 suatu *service connection-oriented* yang meyediakan pengkontrolan aliran data yang sering disebut sebagai *reliability*. Penggabungan kedua protokol ini memungkinkan disediakan layanan yang meluas. TCP/IP adalah protokol *Layer* 3 dan *Layer* 4 dimana internet dibangun.

2.1.5.1 Protocol TCP

Transmission Control Protocol (TCP) adalah sebuah protokol *Layer* 4 yang bersifat *connection-oriented* yang menyediakan transmisi data *full-duplex* yang dapat diandalkan. TCP adalah bagian dari TCP/IP protokol *stack*.

Karakterisitik TCP:

1. Berorientasi sambungan (*connection-oriented*): Sebelum data dapat ditransmisikan antara dua *host,* dua proses yang berjalan pada lapisan aplikasi harus melakukan negosiasi untuk membuat sesi koneksi terlebih dahulu.
2. *Full-duplex*: Untuk setiap *host* TCP, koneksi yang terjadi antara dua *host* terdiri atas dua buah jalur, yakni jalur keluar dan jalur masuk.
3. Dapat diandalkan (*reliable*): Data yang dikirimkan ke sebuah koneksi TCP akan diurutkan dengan sebuah nomor urut paket dan akan mengharapkan paket *positive acknowledgment* dari penerima.
4. *Byte stream*: TCP melihat data yang dikirimkan dan diterima melalui dua jalur masuk dan jalur keluar TCP sebagai sebuah *byte stream* yang berdekatan.
5. Memiliki layanan *flow control*: Untuk mencegah data terlalu banyak dikirimkan pada satu waktu, yang akhirnya membuat "macet" jaringan *inter-network* IP, TCP mengimplementasikan layanan *flow control* yang dimiliki oleh pihak pengirim yang secara terus menerus memantau dan membatasi jumlah data yang dikirimkan pada satu waktu.
6. Melakukan segmentasi terhadap data yang datang dari lapisan aplikasi (dalam DARPA *Reference Model*)

7. Mengirimkan paket secara "*one-to-one*": hal ini karena memang TCP harus membuat sebuah sirkuit logis antara dua buah protokol lapisan aplikasi agar saling dapat berkomunikasi.

2.1.5.2 Internet *Protocol* (IP)

Menurut Sopandi(2010, p63), *Internet Protocol* adalah metode atau protocol untuk melakukan pengalamatan dan *routing* paket data antar *host-host* di jaringan komputer berbasis TCP/IP. Setiap komputer dalam internet setidaknya harus mempunyai sebuah alamat IP yang unik yang mengidentifikasikan komputer tersebut terhadap komputer yang lainnya. Saat ini terdapat standar pengalamatan yang sudah digunakan yaitu IPv4 dengan alamat terdiri dari 32 bit. *Internet Protocol* (IP) juga merupakan *building block* (fondasi) dari internet.

2.1.5.2.1 Pengalamatan IP

IP (*Internet Protocol*) address merupakan bilangan biner 32 bit yang dipisahkan oleh tanda pemisah berupa tanda titik pada setiap 8 bitnya.

IP *Address* terdiri dari 2 bagian yaitu *network* ID dan *host* ID, di mana *network* ID menentukan alamat jaringan, sedangkan *host* ID menentukan alamat *host* atau komputer. Dalam mementukan alamat kelas IP adalah dengan memeriksa 4 bit pertama (bit yang paling kiri) alamat IP.

2.1.5.2.2 Private dan *Public* IP *Address*

1. *Private* IP *Address*

IANA (International Assigned Numbers Authority), mengelompokkan alamat IP *address* yang dinyatakan ***"private",*** artinya hanya untuk digunakan dikalangan sendiri dan tidak berlaku di internet. Alamat IP yang berada di dalam ruangan alamat pribadi dikenal juga dengan alamat pribadi atau *Private Address*. Karena di antara ruangan alamat publik dan ruangan alamat pribadi tidak saling melakukan *overlapping*, maka alamat pribadi tidak akan menduplikasi alamat publik, dan tidak pula sebaliknya.

2. *Public IP Address*

Alamat publik adalah alamat-alamat yang telah ditetapkan oleh Inter NIC dan berisi beberapa buah *network identifier* yang telah dijamin unik. Ketika beberapa alamat publik telah ditetapkan, maka beberapa rute dapat diprogram ke dalam sebuah *router* sehingga *traffic* data yang menuju alamat publik tersebut dapat mencapai lokasinya. Di internet, lalu lintas ke sebuah alamat publik tujuan dapat dicapai, selama masih terkoneksi dengan internet.

2.1.5.2.3 IP *Subnetting*

Sebuah *subnet* memungkinkan arus *traffic* jaringan antara *host* yang akan dipisahkan berdasarkan konfigurasi jaringan.

Dengan mengorganisir *host* ke dalam kelompok logis, *subnetting* dapat meningkatkan keamanan jaringan dan kinerja.

2.1.5.2.4 *Subnet Mask*

Subnet mask adalah istilah teknologi informasi dalam bahasa Inggris yang mengacu kepada angka biner 32 *bit* yang digunakan untuk membedakan *network* ID dengan *host* ID, menunjukkan letak suatu *host*, apakah berada di jaringan *local* atau jaringan luar. *Subnet Mask* (*Extended Netwok Prefix*) bukan sebuah alamat, tetapi menentukan bagian mana dari alamat IP yang merupakan *field Network* dan bagian mana yang merupakan *field Host*.

2.1.6 Routing

Routing merupakan proses penerusan paket data dari suatu alamat sumber (*source network*) ke alamat tujuan (*destination network*) yang dilakukan oleh *router* berdasarkan informasi yang ada pada diri sang *router* itu sendiri, yaitu *routing table*. Jadi *routing table* merupakan alat untuk mengambil keputusan dalam *routing* paket data oleh *router*. *Routing loop* dapat terjadi saat terjadi inkonsistensi dalam tabel *routing* karena konvergensi yang lambat pada saat terjadi perubahan jaringan. *Routing* dapat dilakukan dengan dua cara, yaitu :

- ***Static Route***

  Static route hanya dipakai untuk jaringan kecil, penggunaan *static route* memiliki kelebihan di antaranya tidak mengkonsumsi *resource* cpu *router* (karena keputusan *routing* hanyalah berlandaskan pada isi dari *routing table*), tidak memerlukan *bandwidth* jaringan yang besar, mengingat *router* tidak mengirimkan paket *broadcast*/*multicast* ke *router* tetangganya. Hanya saja karena pengisian entri *routing table*-nya dilakukan manual, rawan akan *human-error* pada saat mengetikkan entri-entrinya.

- ***Dynamic Route***

  Pada *dynamic route*, entri-entri pada *routing table* di *router* dibangun sendiri oleh *router-router* yang berpartisipasi dalam *network* tertentu yang menggunakan *routing protocol* yang sama. Cara ini dipakai jika jaringan memiliki banyak *subnetwork*, dimana jika digunakan cara *static route* tidak efisien bagi administrator jaringan dalam melakukan konfigurasi dan *maintenance router*.

- ***Default Route***

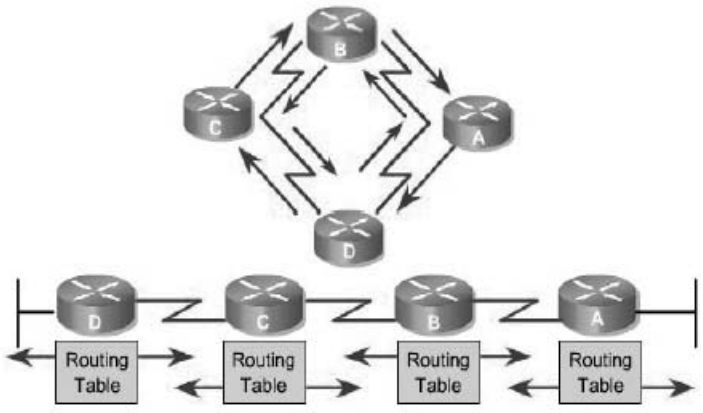
  Default route ini pada dasarnya merupakan *static route* yang memiliki alamat unik, yaitu alamat yang mewakili seluruh jaringan. Secara umum alamat ini adalah 0.0.0.0 dengan subnet mask 255.255.255.255.

2.1.6.1 *Dynamic Routing Protocol*

Dynamic routing protocol terbagi atas tiga kategori:

- ***Distance vector***

Distance vector berarti bahwa *routing protocol* ini dalam menetapkan jalur terbaik (*the best path*) hanya melibatkan jumlah *hop* saja (*hop count*) untuk me-*route* paket data dari satu alamat *network* ke alamat *network* tujuan. Yang termasuk *distance vector* adalah *Routing Information Protocol* (RIP) version 1, RIP version 2, *Interior Gateway routing Protocol* (IGRP).

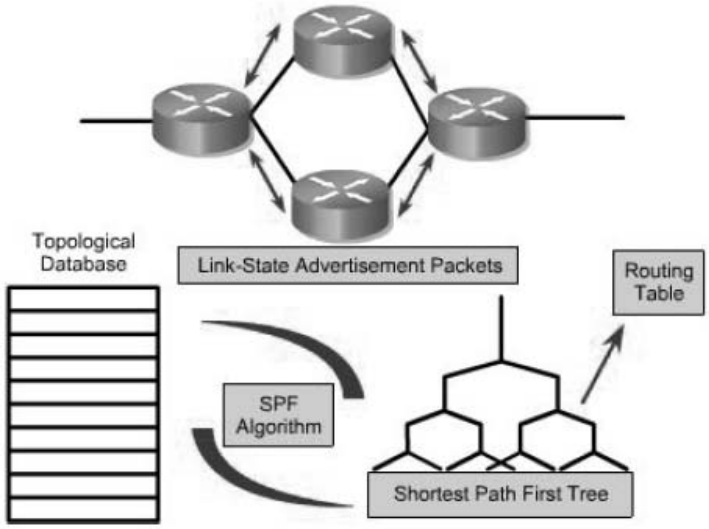


Gambar 2.9 Konsep *Distance Vektor*
(Sumber: http//cnap.binus.ac.id/)

- ***Link State***

Link-state merupakan *routing protocol* yang lebih *modern* dibanding *distance vector*. Algoritma ini menghitung dan menggunakan jalan yang terpendek ke *router* lain, *update* dikirim jika ada perubahan topologi jaringan, lebih cepat untuk *converge*, tidak rentan terhadap *routing loop*, lebih sulit untuk dikonfigurasi,

membutuhkan lebih banyak memori dan *processing power*, lebih sedikit menghabiskan *bandwidth* dibanding *distance vector*, mengambil pandangan umum seluruh topologi jaringan. Yang termasuk *link-state* adalah OSPF, IS-IS.



Gambar 2.10 Konsep *Link-State*
(Sumber: http//cnap.binus.ac.id/)

- ***Hybrid***

  Kategori ini hadir setelah Cisco System membuat *routing protocol* EIGRP (*Enhanced Interior Gateway Routing Protocol*) yang merupakan pengembangan dari IGRP klasik yang bersifat *open standart*. EIGRP cisco ini bersifat *proprietary*, hanya akan berfungsi optimal jika seluruh *device router* yang digunakan bermerek

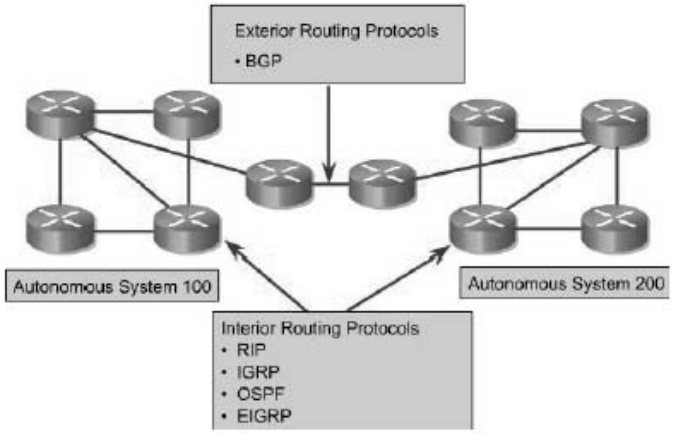
cisco. Kategori ini diklaim memiliki kelebihan yang ada baik pada *Distance Vector* dan juga *Link-State.*

2.1.6.2 Routing Protocol

Dalam suatu jaringan local atau LAN, maka umumnya semua piranti jaringan terhubung dengan satu atau beberapa switch dengan menggunakan kabel LAN. Lain halnya dengan jaringan wireless, piranti wireless adapter terhubung dengan menggunakan frequency radio.

2.1.6.2.1 Autonomous System

AS adalah kumpulan dari jaringan-jaringan yang dalam satu administrasi yang mempunyai strategi routing bersama. AS mungkin dijalankan oleh satu atau lebih operator ketika AS digunakan pada routing ke dunia luar. American Registry of Internet Numbers (ARIN) adalah suatu service provider atau seorang administrator yang memberikan nomor identitas ke AS sebesar 16-bit. Routing protokol seperti Cisco IGRP membutuhkan nomor AS (AS number) yang sifatnya unik.



Gambar 2.11 Struktur AS
(Sumber: http//cnap.binus.ac.id/)

2.1.6.2.2 Routing Information Protocol (RIP)

Routing Information Protocol (RIP) adalah routing vektor jarak-protokol, yang mempekerjakan hop sebagai metrik routing. Palka *down time* adalah 180 detik. Jumlah maksimum hop diperbolehkan untuk RIP adalah 15.

Awalnya setiap router RIP mentransmisikan /menyebarkan pembaruan(*update*) penuh setiap 30 detik. RIP mengimplementasikan *split horizon*, *rute holddown* keracunan dan mekanisme untuk mencegah informasi *routing* yang tidak benar dari yang disebarkan. Ini adalah beberapa fitur stabilitas RIP.

RIP memiliki 3 versi yaitu RIPv1, RIPv2, dan RIPng.

- RIPv1

  RIPv1 didefinisikan pada RFC 1058, dimana menggunakan *classful routing*, tidak menggunakan subnet. Tidak mendukung *Variable Length Subnet Mask* (VLSM).

- RIPv2

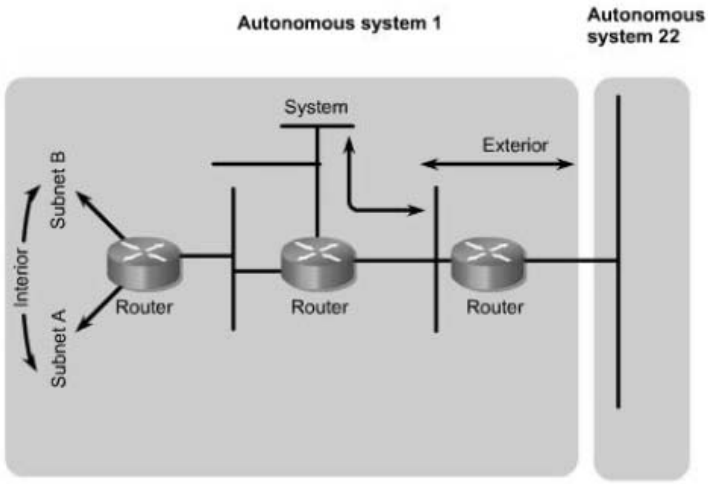
  RIPv2 hadir sekitar tahun 1994, dengan memperbaiki kemampuan akan *Classless Inter-Domain Routing*. Didefinisikan pada RFC 2453.

- RIPng

  RIPng merupakan protokol RIP untuk IPv6. Didefinisikan pada RFC 2080.

2.1.6.2.3 Interior Gateway Routing Protocol (IGRP)

IGRP (Interior Gateway Routing Protocol) adalah juga protocol distance vector yang diciptakan oleh perusahaan Cisco untuk mengatasi kekurangan RIP. Jumlah hop maksimum menjadi 255 dan sebagai metric, IGRP menggunakan bandwidth, MTU, delay dan load. IGRP adalah protocol routing yang menggunakan Autonomous System (AS) yang dapat menentukan routing berdasarkan system, interior atau exterior. Administrative distance untuk IGRP adalah 100.



Gambar 2.12 Jenis-jenis Rute Pada IGRP
(Sumber: http//cnap.binus.ac.id/)

Walaupun IGRP telah memperbaiki sedikit kelemahan pada RIPv1, tetapi IGRP tidak mendukung VLSM dan CIDR. Oleh karena itu, Cisco telah membuat EIGRP untuk memperbaiki masalah ini.

2.1.6.2.4 Enhanced Interior Gateway Routing Protocol **(EIGRP)**

EIGRP adalah lanjutan jarak vektor-*routing* protokol, dengan optimasi untuk meminimalkan *routing* ketidakstabilan yang terjadi setelah perubahan topologi, serta penggunaan dan pengolahan daya *bandwidth* di *router*. Kelemahan utamanya adalah bahwa hal itu hanya berjalan pada peralatan Cisco, yang dapat menyebabkan suatu organisasi yang terkunci

terdalam untuk vendor ini. EIGRP menggunakan beberapa terminologi, yaitu :

1. ***Successor*** : istilah yang digunakan untuk jalur yang digunakan untuk meneruskan paket data.
2. ***Feasible Successor*** : istilah yang digunakan untuk jalur yang akan digunakan untuk meneruskan data apabila *successor* mengalami kerusakan.
3. ***Neighbor table***: istilah yang digunakan untuk tabel yang berisi alamat dan *interface* untuk mengakses ke *router* sebelah
4. ***Topology table***: istilah yang digunakan untuk tabel yang berisi semua tujuan dari *router* sekitarnya.
5. ***Reliable transport protocol***: EIGRP dapat menjamin urutan pengiriman data.

Perangkat EIGRP bertukar informasi *hello packet* untuk memastikan daerah sekitar. Pada *bandwidth* yang besar *router* saling bertukar informasi setiap 5 detik, dan 60 detik pada *bandwidth* yang lebih rendah.

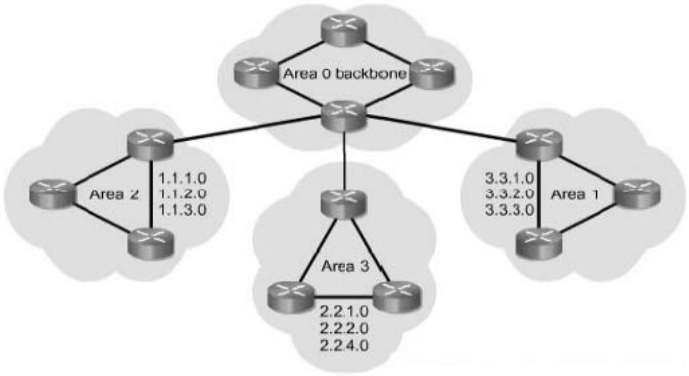
2.1.6.2.5 Open Shortest-Path First (OSPF)

OSPF merupakan *sebuah routing* protokol berjenis IGP yang hanya dapat bekerja dalam jaringan internal suatu ogranisasi atau perusahaan. Selain itu, OSPF juga merupakan *routing* protokol yang berstandar terbuka.

Maksudnya adalah *routing* protokol ini bukan ciptaan dari vendor manapun. Keuntungan utama OSPF dibandingkan RIP adalah OSPF dapat melakukan konvergensi yang cepat dan skalabilitas lebih luas untuk implementasi jaringan yang lebih besar.

Tipe Paket OSPF

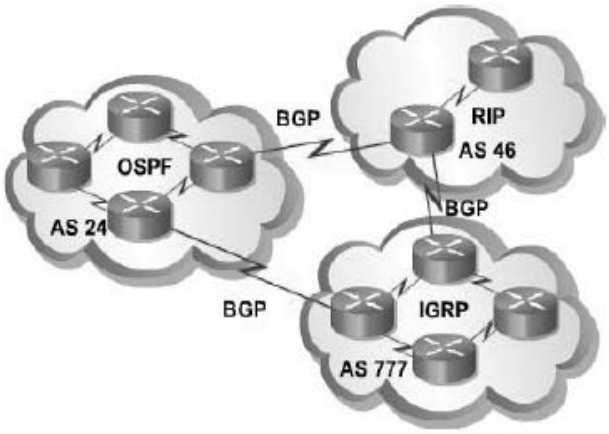
1. Hello – Paket hello digunakan untuk membangun dan memelihara *adjacency* dengan *router* OSPF lainnya.
2. DBD – *Database Description* (DBD) berisi daftar-daftar dari *database link state router* pengirim dan digunakan oleh *router* penerima untuk memeriksa dan dibandingkan dengan *database link state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



Gambar 2.13 Area Pada OSPF
(Sumber: http//cnap.binus.ac.id/)

2.1.6.3.6 *Border Gateway Protocol* (BGP)

Border Gateway Protocol atau lebih familiar dikenal dengan nama **BGP** merupakan sebuah protokol *routing inter-Autonomous System*. Fungsi utama sistem BGP adalah untuk bertukar informasi *network* yang dapat 'dijangkau' (*reachability*) oleh sistem BGP lain, termasuk di dalamnya informasi-informasi yang terdapat dalam list *autonomous system* (AS). BGP berjalan melalui sebuah protokol *transport*, yaitu TCP.



Gambar 2.14 BGP
(Sumber: http//cnap.binus.ac.id/)

2.1.7 Karakteristik Performa Jaringan

2.1.7.1 Paket Loss

Paket Loss adalah kegagalan transmisi paket IP mencapai tujuannya. Kegagalan paket tersebut dapat disebabkan oleh :

1. Terjadinya *overload* trafik di dalam jaringan.
2. Tabrakan (*congestion*) dalam jaringan.
3. *Error* yang terjadi pada media fisik.
4. Kegagalan yang terjadi pada sisi penerima antara lain bisa disebabkan karena *overflow* yang terjadi pada *buffer*.

Di dalam implementasi jaringan IP, nilai *packet loss* ini diharapkan mempunyai nilai minimum.

2.1.7.2 Delay

Delay di dalam jaringan dapat digolongkan sebagai berikut :

- Packetisasi delay

Delay yang disebabkan oleh waktu yang diperlukan untuk proses pembentukan paket IP dari informasi user. Delay ini hanya terjadi sekali saja, yaitu di *source* informasi.

- Queuing delay

  Delay ini disebabkan oleh waktu proses yang diperlukan oleh *router* di dalam menangani transmisi paket di sepanjang jaringan. Umumnya delay ini sangat kecil, kurang lebih sekitar 100 microsecond.

- Delay propagansi

  Proses perjalanan informasi selama di dalam media transmisi, misalnya SDH, coax atau tembaga, menyebabkan delay yang disebut dengan delay propagasi.

2.1.7.3 Throughput

Hal lain yang penting dari jaringan yang dapat diukur secara kuantitatif adalah *throughtput*. *Throughtput* adalah ukuran rata-rata dimana data dapat dikirim melewati jaringan, dan biasanya dispesifikasikan dalam bits per second (bps). Sebagian besar jaringan mempunyai *throughput* sebesar beberapa million bits per second (Mbps), dan sekarang telah mencapai beberapa Gigabits per second (Gbps).

2.2 Teori Khusus

2.2.1 *Multiprotocol Label Switching* (MPLS)

2.2.1.1 Pendahuluan

Menurut Alwayn(2002, p4), ***Multiprotocol Label Switching*** (disingkat menjadi **MPLS**) adalah teknologi arsitektur *network* dimana paket disampaikan pada jaringan *backbone* berkecepatan tinggi. Asas kerjanya menggabungkan beberapa kelebihan dari sistem komunikasi *circuit-switched* dan *packet-switched*. Sebelumnya, paket-paket diteruskan dengan protokol *routing* seperti OSPF, IS-IS, BGP, atau EGP. Protokol *routing* berada pada lapisan *network* dalam *system* OSI, sedangkan MPLS berada di antara lapisan kedua dan ketiga.

Prinsip kerja MPLS ialah menggabungkan kecepatan *switching* pada *layer 2* dengan kemampuan *routing* dan skalabilitas pada *layer 3*. Cara kerjanya adalah dengan menyelipkan *label* di antara *header layer 2* dan *layer 3* pada paket yang diteruskan. Label dihasilkan oleh *Label-Switching Router* dimana bertindak sebagai penghubung jaringan MPLS dengan jaringan luar. *Label* berisi informasi tujuan *node* selanjutnya kemana paket harus dikirim. Kemudian paket diteruskan ke *node* berikutnya, di *node* ini label paket akan dilepas dan diberi label yang baru yang berisi tujuan berikutnya. Paket-paket diteruskan dalam *path* yang disebut LSP (*Label Switching Path*).

2.2.1.2 Komponen MPLS

MPLS terdiri atas sirkuit yang disebut *label-switched path* (LSP) yang menghubungkan *node-node* yang disebut *label-switched router* (LSR). LSR pertama yang merupakan awal tempat masuknya paket disebut dengan *ingress* dan LSR terakhir tempat keluar paket dari MPLS disebut *egress*. Setiap LSP dikaitkan dengan sebuah *forwarding equivalence class* (FEC), yang merupakan kumpulan paket yang menerima perlakukan *forwarding* yang sama di sebuah LSR. FEC diidentifikasikan dengan pemasangan label. Berikut detail komponen yang ada dalam MPLS.

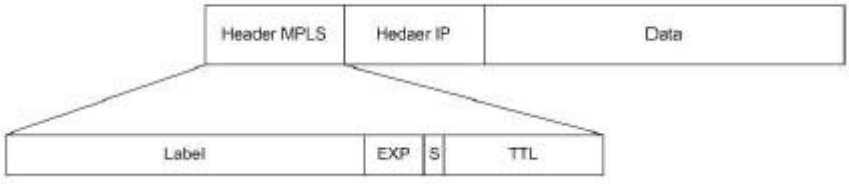
- ***Label Switched Path*** (LSP): Merupakan jalur yang melalui satu atau serangkaian LSR dimana paket diteruskan oleh label *swapping* dari satu MPLS *node* ke MPLS *node* yang lain.
- ***Label Switching Router***: MPLS *node* yang mampu meneruskan paket-paket layer-3
- ***MPLS Edge Node*** atau ***Label Edge Router*** (LER): MPLS *node* yang menghubungkan sebuah MPLS *domain* dengan *node* yang berada diluar MPLS *domain*
- ***MPLS Egress Node***: MPLS *node* yang mengatur trafik saat meninggalkan MPLS *domain*
- ***MPLS ingress Node***: MPLS *node* yang mengatur trafik saat akan memasuki MPLS *domain*
- ***MPLS label***: merupakan label yang ditempatkan sebagai MPLS *header*

- ***MPLS node***: *node* yang menjalankan MPLS. MPLS *node* ini sebagai *control* protokol yang akan meneruskan paket berdasarkan label.

2.2.1.3 MPLS Label

Berbeda dengan ATM yang memecah paket-paket IP, MPLS hanya melakukan enkapsulasi paket IP dengan menempelkan *header* MPLS pada suatu paket. *Header* MPLS terdiri atas 32 bit data, termasuk 20 bit label, 2 bit eksperimen, 1 bit identifikasi *stack*, serta 8 bit TTL. Label adalah bagian dari *header*, memiliki panjang yang bersifat tetap, dan merupakan satu-satunya tanda identifikasi paket. Label digunakan untuk proses forwarding termasuk proses *traffic engineering*.

Berikut pemetaan MPLS *header packet.*



Gambar 2.15 Format MPLS *header* paket
(Sumber: http://www.ittelkom.ac.id/library/)

Gambar diatas merupakan gambar format MPLS *header* paket dengan rincian sebagai berikut.

a. *Label Value* (LABEL)

Merupakan *field* yang terdiri dari 20 bit yang merupakan nilai dari label tersebut.

b. *Experimental Use* (EXP)

Secara teknis *field* ini digunakan untuk keperluan eksperimen. *Field* ini dapat digunakan untuk menangani indikator QoS atau dapat juga merupakan hasil salinan dari bit-bit IP *Precedence* pada paket IP.

c. *Bottom of Stack* (STACK)

Pada sebuah paket terdapat kemungkinan untuk 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*.

d. *Time to Live* (TTL)

Field ini biasanya merupakan hasil salinan dari IP TTL *header*. Nilai *bit* TTL akan berkurang 1 setiap paket melewati hop untuk menghindari terjadinya *packet storms*. Dalam proses pembuatan label ada beberapa metode yang dapat digunakan, yaitu:

- Metode berdasarkan topologi jaringan, yaitu dengan menggunakan protokol IP-*routing* seperti *Open Shortest Path First* (OSPF).
- Metode berdasarkan *resource* suatu paket data, yaitu dengan menggunakan protokol yang dapat mengontrol

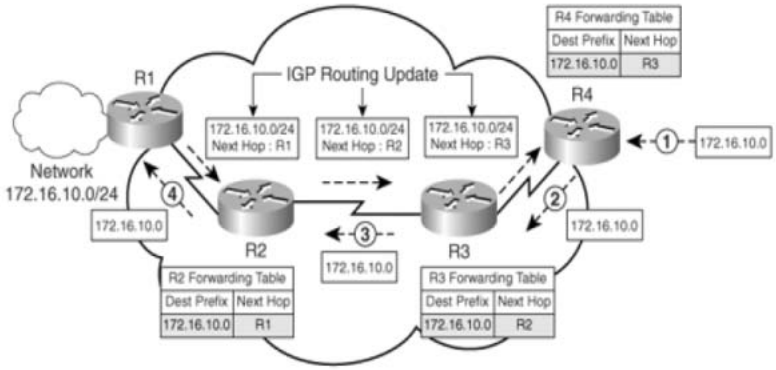
trafik suatu jaringan seperti *Resource Reservation Protocol* (RSVP).

- Metode berdasarkan besar trafik pada suatu jaringan, yaitu dengan menggunakan metode penerimaan paket dalam menentukan tugas dan distribusi suatu label. Setiap LSR memiliki tabel yang disebut *label-switching table*. Tabel itu berisi pemetaan label masuk, label keluar, dan *link* ke LSR berikutnya. Saat LSR menerima paket, label paket akan dibaca, kemudian diganti dengan label keluar, lalu paket dikirimkan ke LSR berikutnya. Selain paket IP, paket MPLS juga bisa dienkapsulasikan kembali dalam paket MPLS. Maka sebuah paket bisa memiliki beberapa *header*, dan bit *stack* pada *header* menunjukan apakah suatu *header* sudah terletak di dasar tumpukan *header* MPLS itu.

2.2.1.4 *Packet Forwarding* pada jaringan IP Tradisional Versus MPLS

Pada jaringan IP tradisional, *routing protocol* digunakan untuk mendistribusikan Layer 3 *routing information*. Proses penerusan paket adalah berdasarkan alamat tujuan. Oleh karena itu, ketika sebuah paket diterima oleh *router*, maka *router* akan mendeterminasikan *next-hop address* menggunakan alamat IP tujuan dengan informasi yang terdapat pada tabel routing. Proses ini akan terus berulang pada tiap loncatan

(*router*) dari sumber ke tujuan.

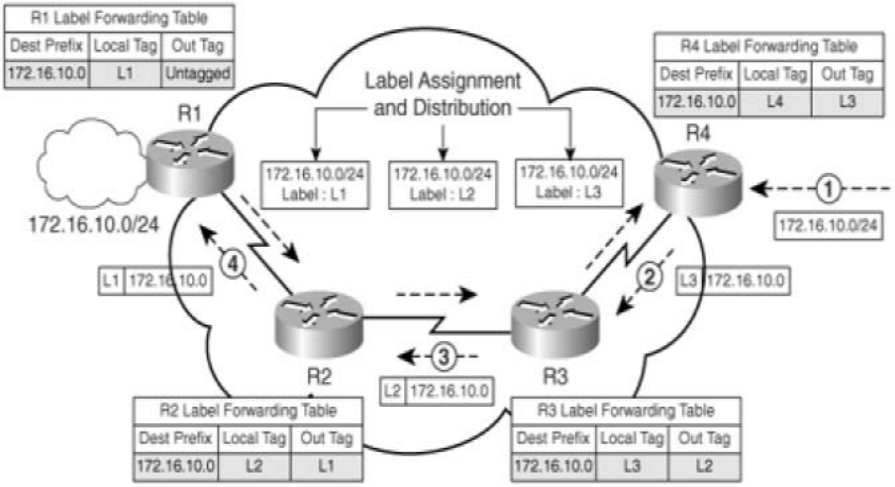


Gambar 2.16 Operasi IP *Forwarding* Tradisional
(Sumber:
http://www.cisco.com/en/US/products/ps6557/prod_presentation_list.html)

Berdasarkan Gambar 2.16 proses penerusan paket adalah sebagai berikut:

1. R4 menerima sebuah paket data yang ditujukan untuk jaringan 172.16.10.0
2. R4 mencari rute untuk jaringan 172.16.10.0 pada label *routing* dan paket diteruskan ke *next-hop*, *router* R3.
3. R3 menerima paket data tersebut dengan tujuan 172.16.10.0 mencari rute untuk jaringan 172.16.10.0. dan meneruskannya ke *router* R2.
4. R2 menerima paket data tersebut dengan tujuan 172.16.10.0 mencari rute untuk jaringan 172.16.10.0. dan meneruskannya ke *router* R1.
5. Karena *router* Rl terhubung langsung ke jaringan 172.16.10.0, Rl akan meneruskan paket tersebut ke *interface* yang tepat.

Sedangkan pada jaringan MPLS, paket data diteruskan berdasarkan label. Label mungkin akan berkoresponden dengan alamat IP tujuan atau dengan parameter lainnya, misalnya kelas-kelas QoS dan alamat sumber.



Gambar 2.17 Operasi Paket *Forwarding* Pada Jaringan MPLS
(Sumber:http://www.cisco.com/en/US/products/ps6557/prod_presentation_list.html)

Berdasarkan Gambar 2.17, proses penerusan paket adalah sebagai berikut :

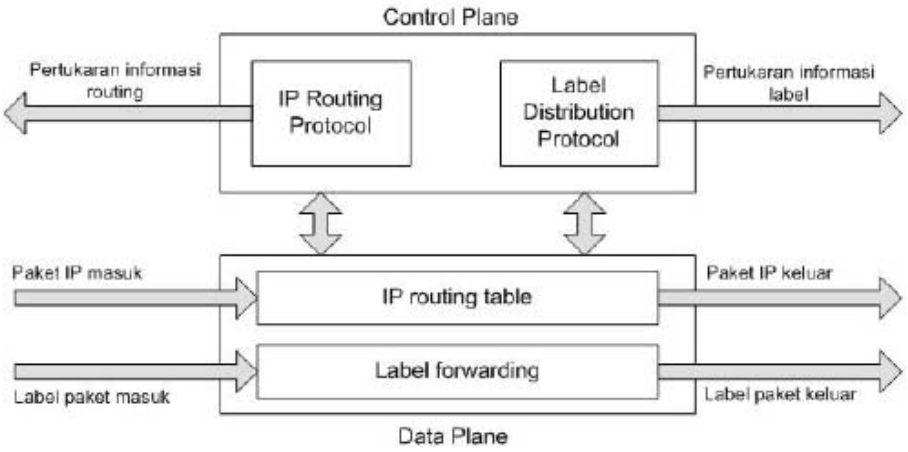
1. R4 menerima sebuah paket data dan jaringan 172.16.10.0 dan mengidentifikasi bahwa rute ke tujuan adalah MPLS *enabled*. Oleh karena itu, R4 meneruskan paket tersebut ke *next-hop router* R3 setelah memakaikan sebuah label L3 pada paket tersebut.
2. R3 menerima *labeled packet* tersebut dengan label L3 dan menukar L3 dengan L2 dan meneruskan paket tersebut ke R2.
3. R2 menerima *labeled packet* tersebut dengan label L2 dan

menukar L2 dengan LI dan meneruskan paket tersebut ke Rl.

4. Rl adalah *border router* di antara jaringan berbasis IP dan MPLS; oleh karena itu, Rl melepaskan label pada paket dan meneruskan paket IP tersebut ke jaringan 172.16.10.0.

2.2.1.5 Arsitektur MPLS

Arsitektur MPLS dirancang guna memenuhi karakteristik-karakteristik yang diharuskan dalam sebuah jaringan kelas *carrier* (pembawa) berskala besar. IETF membentuk kelompok kerja MPLS pada yahun 1997 guna mengembangkan metode umum yang distandarkan. Tujuan dari kelompok kerja MPLS ini adalah untuk menstandarkan protokol-protokol yang menggunakan teknik pengiriman label *swapping* (pertukaran label). Penggunaan label *swapping* ini memiliki banyak keuntungan. Ia bisa memisahkan masalah *routing* dari masukan *forwarding*. *Routing* merupakan masalah jaringan global yang membutuhkan kerjasama dari semua *router* sebagai partisipan. Sedangkan *forwarding* (pengiriman) merupakan masalah setempat. *Router switch* mengambil keputusannya sendiri tentang jalur mana yang akan diambil. MPLS juga memiliki kelebihan yang mampu memperkenalkan kembali *connection stack* ke dalam *dataflow* IP.

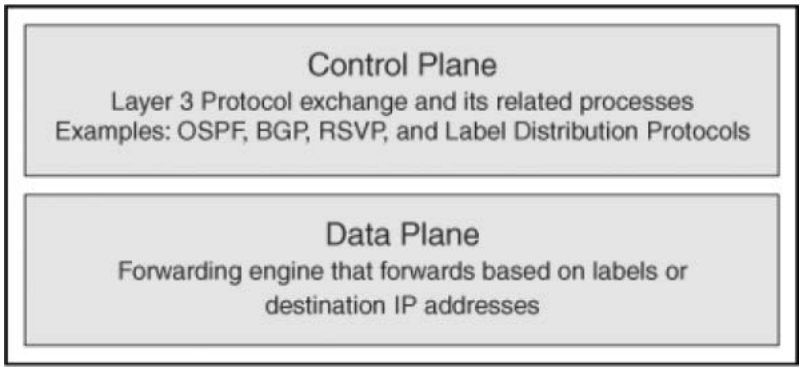


Gambar 2.18 *Control Plane* dan *Data Plane* Pada Router
(Sumber: http://www.ittelkom.ac.id/)

Fungsionalitas MPLS dibagi menjadi dua bagian utama blok arsitektur, yaitu:

1. *Control Plane* - bertanggung jawab dalam hal yang berhubungan dengan pengidentifikasian kemampuan untuk mencapai tujuan. Oleh karena itu. *control plane,* terdiri dari semua informasi pada *Layer* 3. Contoh fungsi *control plane* adalah pertukaran informasi protokol *routing,* seperti OSPF dan BGP. Selain itu, semua fungsi yang berhubungan dengan pertukaran label antara *router-router* tetangga.
2. *Data Plane* - bertugas untuk meneruskan paket-paket data. Paket-paket di sini bisa berarti paket IP *Layer* 3 atau *labeled* IP *packet.* Informasi pada *data plane,* seperti *label values*, adalah berasal dari *control panel.* Pertukaran informasi antara *router-router* tetangga akan memetakan jaringan tujuan ke *labels* pada *control*

plane, yang akan digunakan untuk meneruskan *data plane labeled packet.*



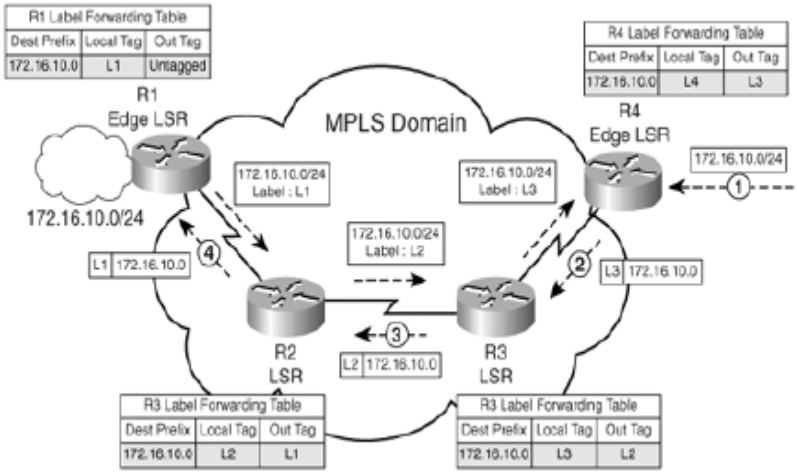
Gambar 2.19 *Control Plane* dan *Data Plane* Pada Router
(Sumber:
http://www.cisco.com/en/US/products/ps6557/prod_presentation_list.html)

2.2.1.6 Istilah-Istilah Dalam MPLS

Beberapa istilah penting dalam MPLS yang akan digunakan terus dalam skripsi ini, yaitu :

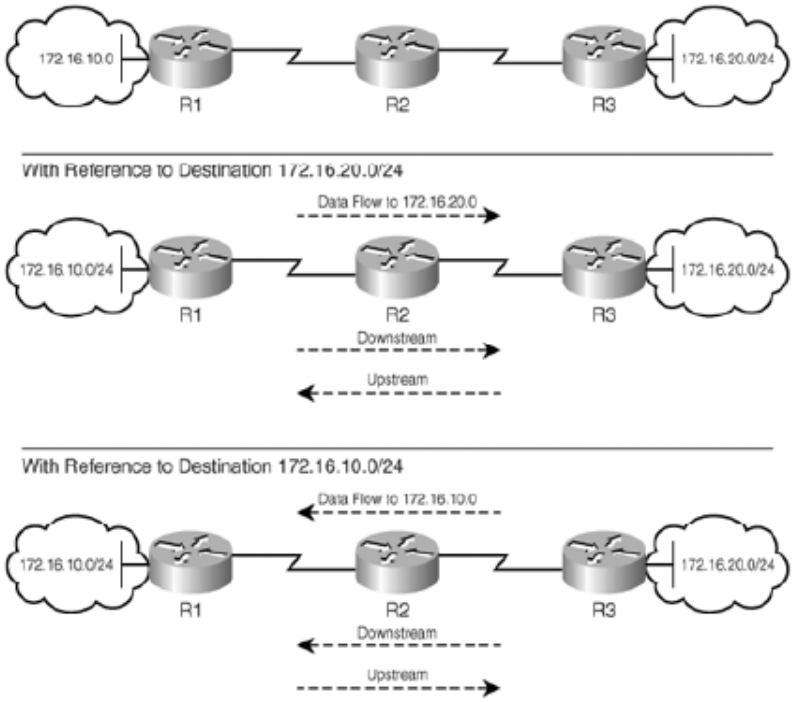
1. *Forwarding Equivalent Class* (FEC) - merupakan sekumpulan paket-paket yang akan mendapatkan perlakuan *forwarding* yang sama (melewati jalur yang sama).
2. MPLS *Label Switch Router* (LSR) - bertugas dalam *label switching*; LSR menerima *labeled packet* dan menukar *label* tersebut dengan *outgoing label* dan meneruskan *labeled packet* baru tersebut dari *interface* yang tepat. Berdasarkan lokasinya dalam *domain* MPLS, LSR bisa bertugas dalam *label imposition* (addition, disebut juga push) atau pun *label disposition* (removal, disebut juga pop).

3. MPLS *Edge-Label Switch Router* (E-LSR) – sebuah LSR pada perbatasan domain MPLS. *Ingress* E-LSR bertugas dalam *label imposition* dan meneruskan paket melalui jaringan *MPLS-enabled*. *Egress* E-LSR bertugas dalam *label disposition* dan meneruskan paket *IP* ke tujuan.



Gambar 2.20 LSR dan E-LSR
(Sumber:
http://www.cisco.com/en/US/products/ps6557/prod_presentation_list.html)

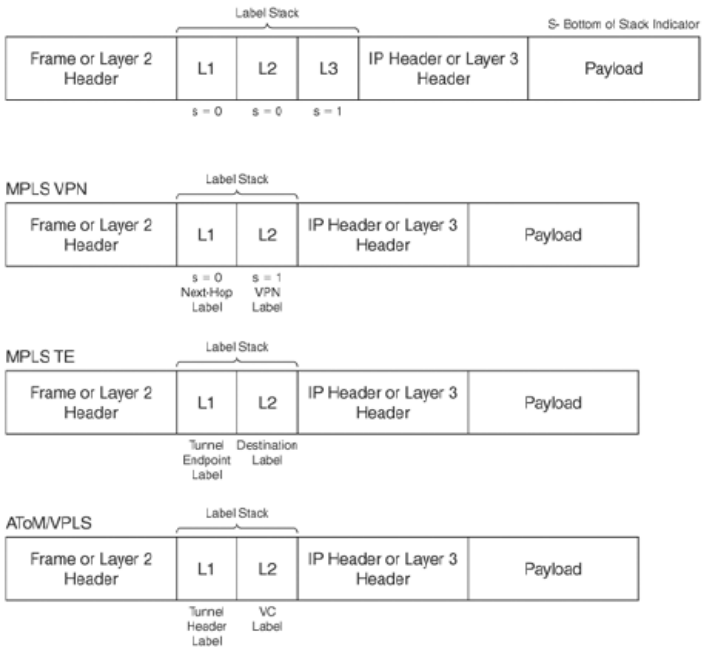
4. MPLS *Label Switched Path* (LSP) – jalur pengiriman paket dari sumber ke tujuan pada jaringan MPLS-*enabled*
5. *Upstream and Downstream* – konsep dari *upstream* dan *downstream* merupakan poros untuk memahami operasi dari distribusi *label* (*control plane*) dan penerusan paket data dalam sebuah *domain* MPLS.



Gambar 2.21 *Upstream* dan *Downstream*
(Sumber: http://www.cisco.com/en/US/products/ps6557/prod_presentation_list.html)

Sebuah *label* MPLS terdiri dari bagian-bagian berikut ini:

1. 20-bit *label value* – nomor yang ditetapkan oleh *router* untuk mengidentifikasikan *prefix* yang diminta.
2. 3-bit *experimental field* – mendefinisikan QoS yang diberikan pada FEC yang telah diberi *label*.
3. 1-bit *bottom-of-stack indicator* – jika E-LSR menambahkan lebih dari satu *label* pada sebuah paket IP, maka akan terbentuk *label stack*. Oleh karena itu, *bottom-of-stack indicator* bertugas untuk mengenal apakah sebuah *label* yang dijumpai merupakan *label* terbawah dalam *label stack*.



Gambar 2.22 MPLS *Label Stack*
(Sumber: http://www.cisco.com/en/US/products/ps6557/prod_presentation_list.html)

4. 8-bit *Time-to-Live field* – memiliki fungsi yang sama dengan IP TTL, di mana paket akan dibuang jika TTL sebuah paket adalah 0. Ketika sebuah *labeled packet* melewati sebuah LSR, nilai TTL-nya akan dikurangi 1.

2.2.2 Traffic Engineering (TE)

2.2.2.1 Pendahuluan

Menurut Ravi Ganesh V, *et al*(2006, p1), *Traffic Engineering* adalah sebuah proses pengontrolan aliran trafik yang melewati jaringan agar kinerja penggunaan *resource* dan jaringan menjadi optimal. Terdapat di dalamnya berupa pemindahan *traffic* sehingga *traffic* dari *link* yang memilki *congestion* dipindahkan ke *link* yang sedang tidak

digunakan. *Traffic Engineering* dapat diimplementasikan dengan cara semudah *tweaking IP metrics* dalam *interface* atau sesuatu yang serumit menjalankan sebuah ATM PVC *full-mesh* dan mengoptimalisasi jalur PVC berdasarkan permintaan *traffic* yang melewatinya.

2.2.2.2 Traffic Engineering pra MPLS

Menurut Osborne, *et al*(p27), *IP traffic engineering* berfungsi mengontrol jalur IP pada jaringan. Namun tidak dapat mengontrol jalur dari mana trafik itu datang, tetapi hanya bisa mengontrol ke mana tujuan trafik itu.

ATM dapat digunakan untuk melewatkan PVC di jaringan dari sebuah sumber trafik ke tujuan, sehingga memiliki lebih banyak hal yang dikontrol aliran trafik dalam jaringan. Beberapa dari ISP terbesar di dunia menggunakan ATM untuk mengendalikan trafik di jaringan. Mereka melakukannya dengan membuat sebuah ATM PVC diantara satu set *router* dan secara periodik mengukur dan menempatkan ulang ATM PVC itu berdasarkan trafik yang diteliti dari *router*. Akan tetapi masalah yang muncul adalah *router* yang bersifat full-mesh menyebabkan O(N2) akan *flooding* ketika suatu hubungan (*link*) mati dan O(N3) *flooding* ketika *router* mati. Hal ini menyebabkan banyak kekhawatiran di beberapa jaringan besar.

2.2.2.3 MPLS dengan Traffic Engineering

Menurut Wastuwibowo(2003, p9), Rekayasa trafik (*traffic engineering*, TE) adalah proses pemilihan saluran data trafik untuk menyeimbangkan beban trafik pada berbagai jalur dan titik dalam *network*. Tujuannya akhirnya adalah memungkinkan operasional *network* yang handal dan efisien, sekaligus mengoptimalkan penggunaan sumber daya dan permormansi trafik. TE untuk MPLS (disebut MPLS-TE) dipandu oleh RFC 2702 (Awduche,1999,3). RFC 2702 menyebutkan tiga masalah dasar berkaitan dengan MPLS-TE, yaitu:

- Pemetaan paket ke dalam FEC
- Pemetaan FEC ke dalam *trunk traffic*
- Pemetaan *trunk traffic* ke topologi *network* fisik melalui LSP

Namun RFC 2702 hanya membahas soal ketiga. Soal lain dikaji sebagai soal-soal QoS. Model MPLS-TE dapat disusun atas komponen-komponen: manajemen *path*, penempatan trafik, penyebaran keadaan *network*, manajemen *network* dan protokol persinyalan.

2.2.2.3.1 Manajemen Path

Manajemen *path* meliputi proses-proses pemilihan *route* eksplisit berdasar kriteria tertentu, serta pembentukan dan pemeliharaan *tunnel* LSP dengan aturan-aturan tertentu. Proses pemilihan *route* dapat dilakukan secara administrative, atau secara otomatis dengan proses *routing* yang bersifat *constraint-*

based. Tujuannya adalah untuk mengurangi pekerjaan manual dalam TE.

2.2.2.3.2 Penempatan Trafik

Setelah LSP dibentuk, trafik harus dikirimkan melalui LSP. Manajemen trafik berfungsi mengalokasikan trafik ke dalam LSP yang telah dibentuk. Ini meliputi fungsi pemisahan, yang membagi trafik atas kelas-kelas tertentu, dan fungsi pengiriman, yang memetakan trafik itu ke dalam LSP.

2.2.2.3.3 Penyebaran Informasi Keadaan Network

Penyebaran ini bertujuan membagi informasi topologi *network* ke seluruh LSR di dalam *network*. Ini dilakukan dengan *protocol gateway* seperti IGP yang telah diperluas. Perluasan informasi meliputi *bandwith* link maksimal, alokasi trafik maksimal, pengukuran TE *default, bandwith* yang dicadangkan untuk setiap kelas prioritas, dan atribut-atribut kelas resource. Informasi-informasi ini akan diperlukan oleh *protocol* persinyalan untuk memilih *routing* yang paling tepat dalam pembentukan LSP.

2.2.2.3.4 Manajemen Network

Performansi MPLS-TE tergantung pada kemudahan mengukur dan mengendalikan *network*. Manajemen *network*

meliputi konfigurasi *network*, pengukuran *network*, dan penanganan kegagalan *network*. Pengukuran terhadap LSP dapat dilakukan seperti pada paket data lainnya. *Traffic flow* dapat diukur dengan melakukan *monitoring* dan menampilkan statistika hasilnya. *Path loss* dapat diukur dengan melakukan *monitoring* pada ujung-ujung LSP, dan mencatat trafik yang hilang. *Path delay* dapat diukur dengan mengirimkan paket *probe* menyeberangi LSP, dan mengukur waktunya. Notifikasi dan *alarm* dapat dibangkitkan jika parameter-parameter yang ditentukan itu telah melebihi ambang batas.

2.2.2.3.5 Protokol Persinyalan

Pemilihan *path*, sebagai bagian dari MPLS-TE, dapat dilakukan dengan dua cara: secara manual oleh administrator, atau secara otomatis oleh suatu protocol persinyalan. Dua protocol persinyalan yang umum digunakan untuk MPLS-TE adalah CR-LDP dan RSVP-TE.

RSVP-TE memperluas *protocol* RSVP yang sebelumnya telah digunakan untuk IP, untuk mendukung distribusi label dan *routing* eksplisit. Sementara itu CR-LDP memperluas LDP yang sengaja dibuat untuk distribusi label, agar dapat mendukung persinyalan berdasar QoS dan *routing* eksplisit.

Ada banyak kesamaan antara CR-LDP dan RSVP-TE dalam kalkulasi *routing* yang bersifat *constraint-based*.

Keduanya menggunakan informasi QoS yang sama untuk menyusun *routing* eksplisit yang sama dengan alokasi *resource* yang sama. Perbedaan utamanya adalah dalam meletakkan layer tempat *protocol* persinyalan bekerja. CR-LDP adalah *protocol* yang bekerja di atas TCP atau UDP, sedangkan RSVP-TE bekerja langsung di atas IP. Perbandingan kedua *protocol* ini dipaparkan dalam table berikut.

Tabel 2.1 Tabel Perbandingan CR-LDP dengan RSVP-TE

Feature	CR-LDP	RSVP-TE
Transport	TCP and UDP	Raw IP
Security	IP-Sec	RSVP Authentication
Multipoint-to-point	Yes	Yes
LSP merging	Yes	Yes
LSP state	Hard	Soft
LSP refresh	Not needed	Periodic
Redundancy	Hard	Easy
Rerouting	Yes	Yes
Explicit routing	Strict and loose	Strict and loose
Route pinning	Yes	By recording path
LSP pre-emption	Priority based	Priority based
LSP protection	Yes	Yes
Shared reservations	No	Yes
Traffic control	Forward path	Reverse path
Policy control	Implicit	Explicit
Layer 3 protocol ID	No	Yes