

BAB II

TINJAUAN PUSTAKA

2.1. MPLS (*Multi Protocol Label Switching*)

Seiring dengan kemajuan teknologi informasi dan telekomunikasi, maka kebutuhan terhadap suatu jaringan akan semakin meningkat, terutama untuk menghubungkan jaringan yang satu dengan jaringan yang lain, dimana kedua tempat jaringan tersebut letaknya saling berjauhan, maka untuk menghubungkan keduanya agar terjadi suatu koneksi yang lebih cepat dan lebih baik maka diperlukan suatu jalur yang dinamakan *Multi Protocol Label Switching* (MPLS).

MPLS adalah suatu teknologi penyampaian paket pada jaringan berkecepatan tinggi yang menggabungkan beberapa kelebihan dari sistem komunikasi circuit-switched dan packet switched yang melahirkan teknologi yang lebih baik dari keduanya [3]. MPLS bekerja pada packets dengan MPLS header, yang berisi satu atau lebih label. Header MPLS terdiri atas 32 bit data, termasuk 20 bit label, 2 bit eksperimen, dan 1 bit identifikasi stack, serta 8 bit TTL. Label pada MPLS digunakan untuk proses forwarding, termasuk proses traffic engineering.

Diharapkan dengan adanya jalur MPLS tersebut maka suatu jaringan dapat terhubung dan terkoneksi dengan mudah dan diharapkan proses pengaksesannya bisa lebih cepat dan lebih baik.

2.1.1. Pengertian MPLS

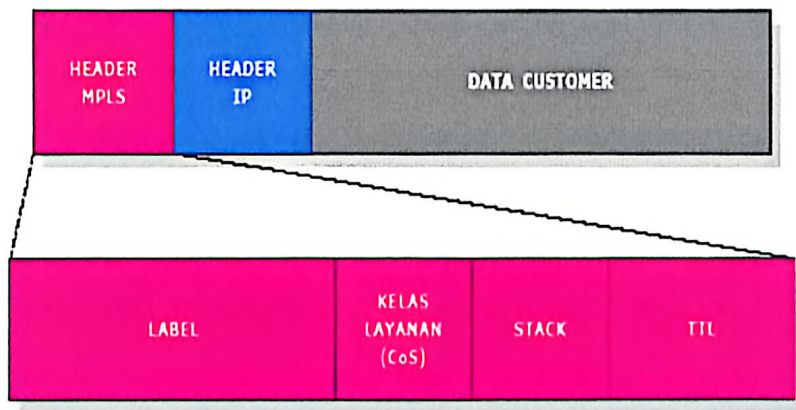
Multiprotocol Label Switching (MPLS) [2] adalah arsitektur network yang didefinisikan oleh IETF untuk memadukan mekanisme label swapping di layer 2 dengan routing di layer 3 untuk mempercepat pengiriman paket.

Paket-paket pada MPLS diteruskan dengan protokol routing seperti OSPF, BGP atau EGP. Protokol routing berada pada layer 3 sistem OSI, sedangkan MPLS berada di antara layer 2 dan 3. OSPF (Open Shortest Path First) adalah routing protocol berbasis link state (dilihat dari total jarak) setelah antar router bertukar informasi maka akan terbentuk database pada masing – masing router.

BGP (Border Gateway Protocol) adalah router untuk jaringan external yang digunakan untuk menghindari routing loop pada jaringan internet.

2.1.2. Enkapsulasi Paket

Tidak seperti ATM yang memecah paket-paket IP, MPLS hanya melakukan enkapsulasi paket IP, dengan memasang header MPLS. Header MPLS terdiri atas 32 bit data, termasuk 20 bit label, 2 bit eksperimen, dan 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. Untuk mengetahui enkapsulasi paket pada MPLS dapat dilihat pada Gambar 2.2.



Gambar 2.1. Enkapsulasi paket pada MPLS

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 menunjukkan apakah suatu header sudah terletak di 'dasar' tumpukan header MPLS itu. Adapun bagian dan komponen MPLS yaitu,

a. LSRs dan LERs

LSR adalah alat penerus kecepatan tinggi dalam inti dari suatu jaringan MPLS yang menggunakan protokol pemberian isyarat label sesuai dan

kecepatan tinggi menswitch data yang didasarkan alur yang telah dibentuk. LER adalah suatu alat yang beroperasi di jaringan akses dan MPLS. LERs mendukung berbagai port yang dihubungkan ke network(seperti penyiaran ulang, ATM dan Ethernet).

b. FEC

Sebagai lawan IP konvensional dalam MPLS, tugas dari FEC dilakukan hanya sekali ketika paket masuk jaringan itu. FECs didasarkan pada kebutuhan jasa atau pelayanan yang ditentukan ke dalam satuan paket. Masing-Masing LSR membangun suatu tempat untuk menetapkan suatu Label Information Base (LIB) apakah terdiri atas FEC.

c. Labels and Label Findings

Suatu label dalam format yang paling sederhana berguna untuk mengidentifikasikan alur suatu paket. Label ini memberikan batasan batasan sebagai berikut,

- tujuan unicast routing
- teknik traffic
- multicast
- virtual private network (VPN) – QoS

d. Label Creation

Ada beberapa metode yang digunakan di dalam penciptaan label yaitu :

- metode topology, dengan menggunakan proses normal dari routing protokol seperti OSPF dan BGP
- metode request, dengan menggunakan proses yang berdasarkan control traffic seperti RSVP
- metode traffic, dengan menggunakan penerimaan paket ke penyaluran trigger dari label

e. Label Distribution

Protokol yang ada, seperti BGP, digunakan sebagai informasi label dalam protokol itu. IETF juga menggambarkan suatu protokol baru yang dikenal sebagai distribusi label protokol karena pemberian isyarat yang tegas dan manajemen ruang. Suatu ringkasan dari berbagai rencana untuk pertukaran label sebagai berikut:

- LDP, IP ditujukan ke dalam label
- RSVP, CR-LDP digunakan untuk reservasi sumber daya dan teknik traffic.
- PIM (PROTOCOL Multicast), digunakan sendiri untuk multicast label negara yang memetakan.
- BGP, eksternal label (VPN)

f. Label Switched Paths (LSPs)

Di dalam suatu daerah MPLS, suatu alur disediakan paket yang ditentukan untuk bepergian didasarkan pada suatu FEC. LSP disediakan sebelum transmisi data. MPLS menyediakan dua pilihan berikut untuk menyediakan suatu LSP :

- *hop-by-hop routing*, setiap LSR dengan bebas memilih loncatan berikutnya untuk FEC ditentukan.
- *explicit routing*, seperti ke sumber routing.

g. Label Spaces

Label yang digunakan oleh suatu LSR untuk FEC-Label binding digolongkan sebagai berikut:

- per platform, Label-label dialokasikan dari suatu common pool. Tidak ada dua label yang didistribusikan ke antarruang yang berbeda yang mempunyai harga sama.
- per interface, jangkauan label disesuaikan dengan antar ruang. Nilai-Nilai label menyajikan tentang alat penghubung yang berbeda bisa sama.

h. Label Merging

Arus traffic yang datang dari alat penghubung berbeda dapat digabungkan bersama-sama dan yang diswitch menggunakan suatu label umum jika mereka sedang melintasi jaringan ke arah tujuan akhir sama. Ini dikenal sebagai stream merging.

i. Label Retention

MPLS menggambarkan label bindings diterima dari LSRS bukanlah loncatan berikutnya untuk FEC yang ditentukan.

j. Label Control

MPLS menggambarkan gaya untuk mendistribusikan label ke LSRs yang berdekatan.

- independent, suatu LSR mengenali FEC tertentu dan membuat keputusan untuk mengikat suatu label kepada FEC dengan bebas untuk mendistribusikannya. FECs baru dikenali di mana saja rute baru yang kelihatan oleh penerus.
- ordered, suatu LSR mengikat suatu label untuk FEC tertentu dan hanya untuk penerus jalan ke luar atau telah menerima suatu label yang mengikat untuk FEC dari loncatan LSR berikutnya. Gaya ini direkomendasikan untuk ATM-LSRs.

k. Signaling Mechanism

- label request, menggunakan mekanisme ini, suatu LSR meminta suatu label dari nya ke downstream neighbor sehingga dapat mengikat FEC yang spesifik. Mekanisme ini dapat digunakan selama rantai LSRs yang atas sampai ke luar LER.
- label mapping, respon ke table request , suatu ke downstream LSR akan mengirimkan suatu label kepada ke pemrakarsa upstream yang menggunakan label yang memetakan mekanisme.

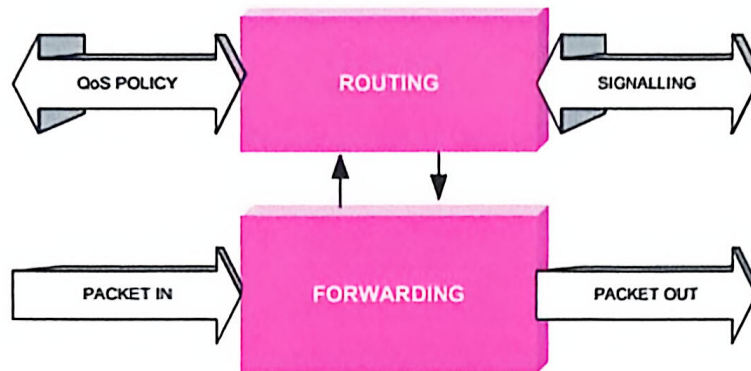
l. Label Distribution Protocol

LDP adalah suatu protokol baru untuk distribusi label yang mengikat informasi ke LSRs di dalam suatu jaringan MPLS. LDP digunakan untuk peta FECs ke label, pada gilirannya membuat LSPs

2.1.3. Arsitektur MPLS

MPLS (*Multi-Protocol Label Switching*), adalah arsitektur network yang didefinisikan oleh IETF untuk memadukan mekanisme label swapping di layer 2 dengan routing di layer 3 untuk mempercepat pengiriman **paket[2]**. Network MPLS terdiri atas sirkit yang disebut label-switched path (LSP), yang menghubungkan titik-titik yang disebut label-switched router (LSR). Setiap LSP dikaitkan dengan sebuah forwarding equivalence class (FEC), yang merupakan

kumpulan paket yang menerima perlakuan forwarding yang sama di sebuah LSR. FEC diidentifikasi dengan pemasangan label.



Gambar 2.2. Arsitektur MPLS

Untuk membentuk LSP, diperlukan suatu protokol persinyalan. Protokol ini menentukan *forwarding* berdasarkan label pada paket. Label yang pendek dan berukuran tetap mempercepat proses forwarding dan mempertinggi fleksibilitas pemilihan path. Hasilnya adalah network datagram yang bersifat lebih *connection-oriented*[3].

1. Sisi LSR pertama:

- Label imposition: Penggolongan & Pelabelan pada

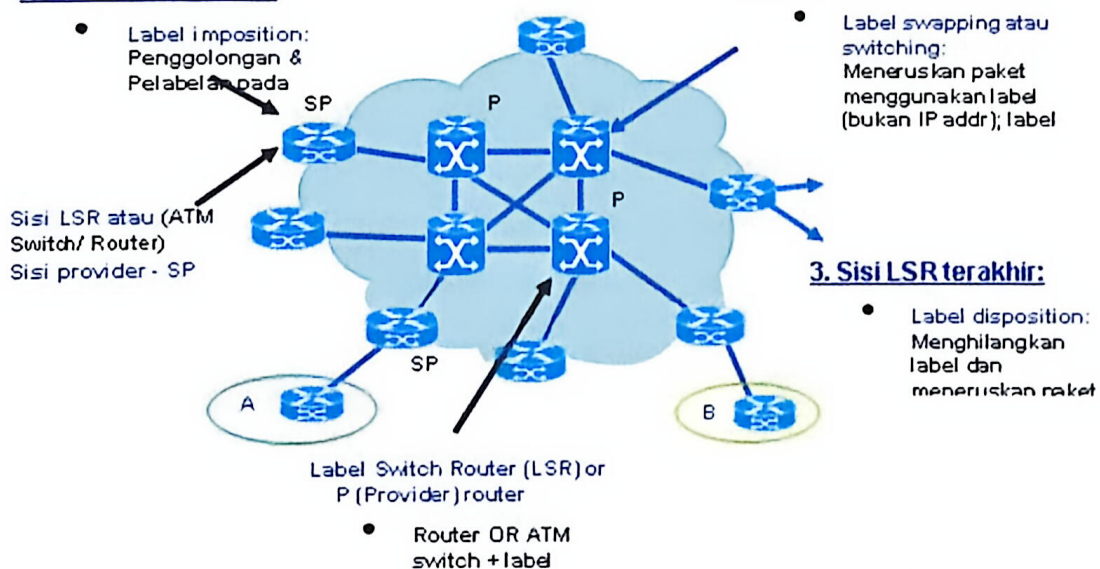
Sisi LSR atau (ATM Switch/ Router)
Sisi provider - SP

2. Inti:

- Label swapping atau switching: Meneruskan paket menggunakan label (bukan IP addr); label

3. Sisi LSR terakhir:

- Label disposition: Menghilangkan label dan meneruskan paket



Gambar 2.3 Arsitektur Jaringan MPLS

- 1) Penggolongan dan pemberian label pada packet. Setelah itu packets akan menuju provider (P). Dari provider, packet akan diteruskan ke inti.

- 2) Pada inti, packet diteruskan berdasarkan label bukan berdasarkan pada IP address. Label ini menunjukkan penggolongan class (A, B, C, D) dan tujuannya.
- 3) Menghilangkan label dan meneruskan packet pada sisi penerima.

2.1.4. Struktur Jaringan MPLS

Struktur jaringan MPLS terdiri dari edge Label Switching Routers atau edge LSRs yang mengelilingi sebuah core Label Switching Routers (LSRs)[7]. Adapun elemen-elemen dasar penyusun jaringan MPLS ialah :

- **Edge Label Switching Routers (ELSR)**

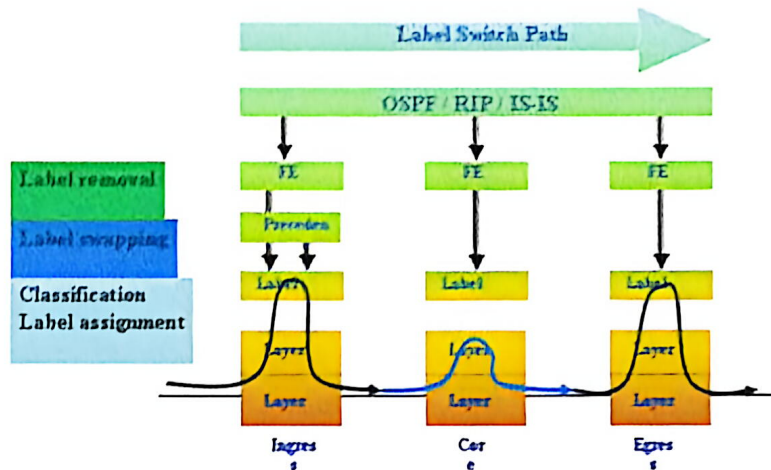
Edge Label Switching Routers ini terletak pada perbatasan jaringan MPLS, dan berfungsi untuk mengaplikasikan label ke dalam paket-paket yang masuk ke dalam jaringan MPLS. Sebuah MPLS Edge Router akan menganalisa header IP dan akan menentukan label yang tepat untuk dienkapsulasi ke dalam paket tersebut ketika sebuah paket IP masuk ke dalam jaringan MPLS. Dan ketika paket yang berlabel meninggalkan jaringan MPLS, maka Edge Router yang lain akan menghilangkan label tersebut. Label Switches. Perangkat Label Switches ini berfungsi untuk menswitch paket-paket ataupun sel-sel yang telah dilabeli berdasarkan label tersebut. Label Switches ini juga mendukung Layer 3 routing ataupun Layer 2 switching untuk ditambahkan dalam label switching. Operasi dalam label switches memiliki persamaan dengan teknik switching yang biasa dikerjakan dalam ATM.

- **Label Distribution Protocol (LDP)**

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

2.1.5. Proses pada MPLS

Untuk mengetahui prinsip kerja yang terjadi pada MPLS ditunjukkan pada Gambar 2.4.



Gambar 2.4. Prinsip Kerja pada Jaringan MPLS

1. Prinsip kerja MPLS ialah menggabungkan kecepatan switching pada layer 2 dengan kemampuan routing dan skalabilitas pada layer 3.
2. Cara kerjanya adalah dengan menyelipkan label di antara header layer 2 dan 3 pada paket yang diteruskan.
3. Label dihasilkan oleh Label-Switching Router dimana bertindak sebagai penghubung jaringan MPLS dengan jaringan luar.
4. 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.
5. Paket-paket diteruskan dalam path yang disebut LSP (*Label Switching Path*).

2.1.6. Standarisasi Protokol MPLS

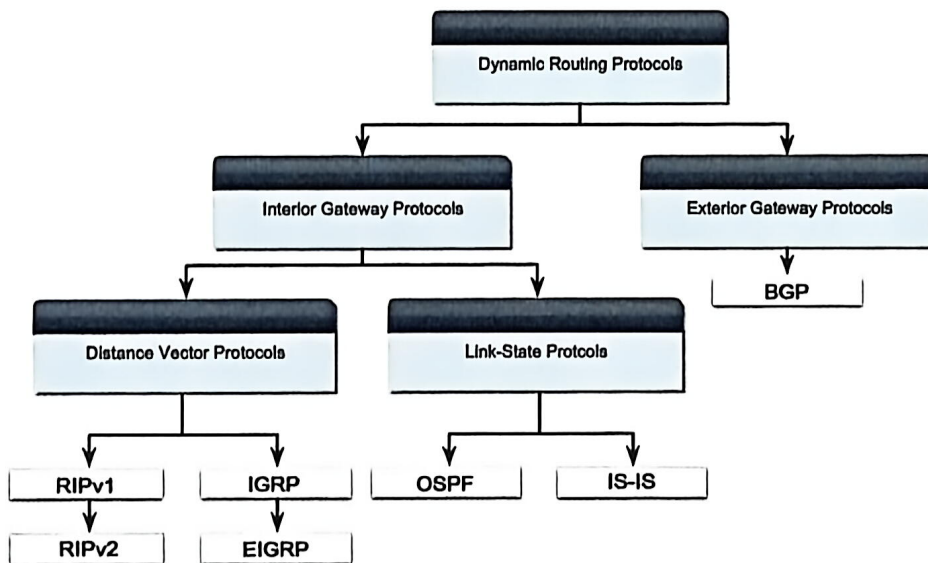
Ada dua standarisasi protokol untuk manage alur MPLS yaitu :

- a) CR-LDP (Constraint-based Routing Label Distribution Protocol)
- b) RSVP-TE, suatu perluasan protocol RSVP untuk traffic rancang bangun

- Suatu header MPLS tidak mengidentifikasi jenis data yang dibawa pada alur MPLS.
- Jika header membawa 2 tipe jalur yang berbeda diantara 2 router yang sama, dengan treatment yang berbeda dari masing – masing jenis core router, maka header MPLS harus menetapkan jalurnya untuk masing – masing jenis traffic

2.2. Routing Protocol

Routing IP adalah proses pengiriman data dari satu host dalam satu network ke host dalam network yang lain melalui suatu router[6]. Agar router dapat mengetahui bagaimana meneruskan paket paket ke alamat yang dituju dengan menggunakan jalur terbaik, router menggunakan peta atau tabel routing. Table routing adalah table yang memuat seluruh informasi IP address dari interfaces router yang lain sehingga router yang satu dengan router lainnya bisa berkomunikasi.



Gambar 2.5. *Routing Protocol* pada MPLS

Routing table hanya memberikan informasi sedang routing algoritma yang menganalisa dan mengatur routing table. Intinya, router hanya tahu cara menghubungkan network atau subnet yang terhubung langsung dengan router tersebut. Router berdasarkan cara pemetaan / routing dibagi tiga:

- 1) *Static Routing*
- 2) *Default Routing*
- 3) *Dynamic Routing*

2.2.1. *Dynamic Routing*

Dalam *Dynamic routing*, *Protocol Routing* menjalankan tugasnya secara aktif tanpa menunggu instruksi dari network admin. Terdapat beberapa jenis routing protocol yang termasuk dalam dynamic routing protocol. Semua routing protokol bertujuan mencari rute tersingkat untuk mencapai tujuan. Dan masing-masing protokol mempunyai cara dan metodenya sendiri-sendiri. Secara garis besar, routing protokol dibagi menjadi Interior Routing Protocol dan Exterior Routing Protocol. Keduanya akan diterangkan sebagai berikut :

A. *Interior Routing Protocol*

Sesuai namanya, interior berarti bagian dalam. Dan interior routing protocol digunakan dalam sebuah network yang dinamakan *autonomous systems* (AS). AS dapat diartikan sebagai sebuah network (bisa besar atau pun kecil) yang berada dalam satu kendali teknik. AS bisa terdiri dari beberapa sub network yang masing-masingnya mempunyai gateway untuk saling berhubungan. Interior routing protocol mempunyai beberapa macam implementasi protokol, yaitu :

- ***RIP (Routing Information Protocol)***

Merupakan protokol routing yang paling umum dijumpai karena biasanya sudah included dalam sebuah sistem operasi, biasanya unix atau novell. RIP memakai metode distance-vector algoritma. Algoritma ini bekerja dengan menambahkan satu angka metrik kepada routing apabila melewati satu gateway. Satu kali data melewati satu gateway maka angka metriknya bertambah satu (atau dengan kata lain naik satu hop). RIP hanya bisa menangani 15 hop, jika lebih maka host tujuan dianggap tidak dapat dijangkau.

Oleh karena alasan tadi maka RIP tidak mungkin untuk diterapkan di sebuah AS yang besar. Selain itu RIP juga mempunyai kekurangan dalam hal network masking. Namun kabar baiknya, implementasi RIP

tidak terlalu sulit jika dibandingkan dengan OSPF yang akan diterangkan berikut ini.

- **OSPF (*Open Shortest Path First*)**

Merupakan protokol routing yang kompleks dan memakan resource komputer. Dengan protokol ini, route dapat dibagi menjadi beberapa jalan. Maksudnya untuk mencapai host tujuan dimungkinkan untuk mencapainya melalui dua atau lebih rute secara paralel.

- ***Intermediate System-to-Intermediate System (IS-IS)***

Intermediate System-to-Intermediate System (IS-IS) adalah protokol yang besar digunakan oleh perangkat jaringan untuk menentukan cara terbaik untuk datagram dipromosikan dari sisi ke sisi paket switched jaringan dan proses ini disebut routing[6]. Ini didefinisikan dalam ISO / IEC 10589 2002 dalam desain referensi OSI. Menengah sistem-ke-intermediate sistem (IS-IS) membedakan antara tingkat-tingkat seperti tingkat 1 dan tingkat 2. Protokol routing dapat diubah tanpa perlu menghubungkan wilayah intra routing protokol.

B. *Exterior Routing Protocol*

AS merupakan sebuah network dengan sistem policy yang pegang dalam satu pusat kendali. Internet terdiri dari ribuan AS yang saling terhubung. Untuk bisa saling berhubungan antara AS, maka tiap-tiap AS menggunakan exterior protocol untuk pertukaran informasi routingnya. Informasi routing yang dipertukarkan bernama *reachability* information (informasi keterjangkauan). Tidak banyak router yang menjalankan routing protokol ini. Hanya router utama dari sebuah AS yang menjalankannya. Dan untuk terhubung ke internet setiap AS harus mempunyai nomor sendiri. Protokol yang mengimplementasikan exterior :

- **EGP (*Exterior Gateway Protocol*)**

Protokol ini mengumumkan ke AS lainnya tentang network yang berada di bawahnya. Pengumumannya kira-kira berbunyi : " Kalau hendak pergi ke AS nomor sekian dengan nomor network sekian, maka silahkan melewati saya" . Router utama menerima routing dari router-router AS yang lain tanpa mengevaluasinya. Maksudnya, rute untuk ke

sebuah AS bisa jadi lebih dari satu rute dan EGP menerima semuanya tanpa mempertimbangkan rute terbaik.

- **BGP (*Border Gateway Protocol*)**

BGP sudah mempertimbangkan rute terbaik untuk dipilih. Seperti EGP, BGP juga mempertukarkan reachability information. Fungsi utama dari routing protocol adalah mencari dan mengumpulkan informasi tentang hal-hal yang diperlukan dalam proses transmisi data antar jaringan melalui router, informasi tersebut kemudian disimpan dalam routing tabel yang nantinya dipakai dalam kegiatan routing. Secara umum routing protocol terbagi atas static dan dynamic routing, static routing hampir tidak memiliki pembagian lain. Static routing hanya bergantung dari update secara manual yang dilakukan oleh administrator dari router.

2.3.2 *Metric*

Setiap protokol *routing* memiliki parameter untuk menentukan pemilihan jalur terbaik, parameter ini disebut dengan *metric*. Adapun jenis *metric* yang digunakan oleh tiap-tiap *protocol routing* adalah sebagai berikut :

- 1) *Hop Count* : Metric yang menggunakan hop count akan menghitung jumlah lompatan router yang dilalui nya hingga mencapai remote network, jumlah hop terkecil akan menjadi pemilihan jalur terbaik.
- 2) *Cost* : Metric yang menggunakan harga (cost) pada setiap link yang ada didalam jaringan, cost dengan nilai terkecil akan dipilih menjadi jalur terbaik.
- 3) *Bandwidth* : Metric yang menggunakan bandwidth sebagai pencarian jalur terbaik, bandwidth dengan kapasitas yang besar akan digunakan sebagai jalur terbaik.
- 4) *Load* : Metric yang menggunakan perhitungan beban setiap path, beban terkecil akan dijadikan sebagai jalur terbaik yang akan dipilih.
- 5) *Delay* : Metric yang menggunakan waktu untuk mengirimkan paket data dari setiap path, nilai delay terkecil akan dijadikan sebagai jalur terbaik yang akan dipilih.

- 6) *Reliability* : Metric yang menggunakan nilai kehandalan dari sebuah path, nilai reliability tertinggi akan dijadikan sebagai jalur terbaik yang akan dipilih.

Secara default tiap masing-masing protocol routing mempunyai metric sebagai berikut :

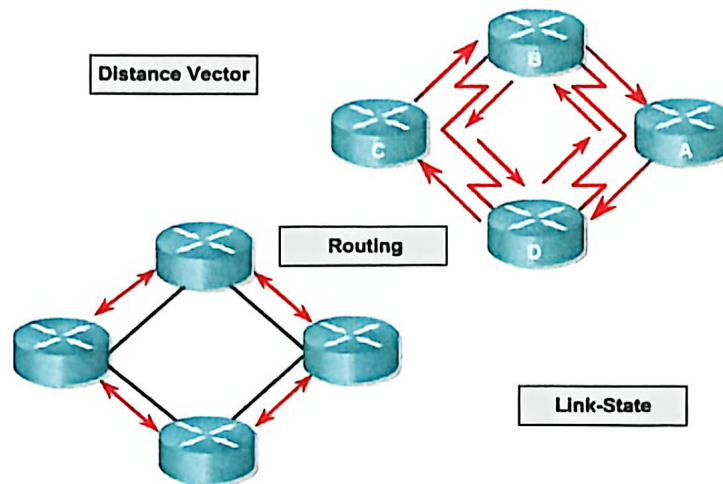
Tabel 2.1. Routing Berdasarkan Metric

<i>Protocol Routing</i>	<i>Metric</i>
RIP	<i>Hop Count</i>
IGRP dan EGRP	Gabungan <i>Bandwidth, Load, Delay</i> , dan <i>Reliability</i>
OSPF dan IS-IS	<i>Cost</i>

2.3. Metode Routing IS-IS

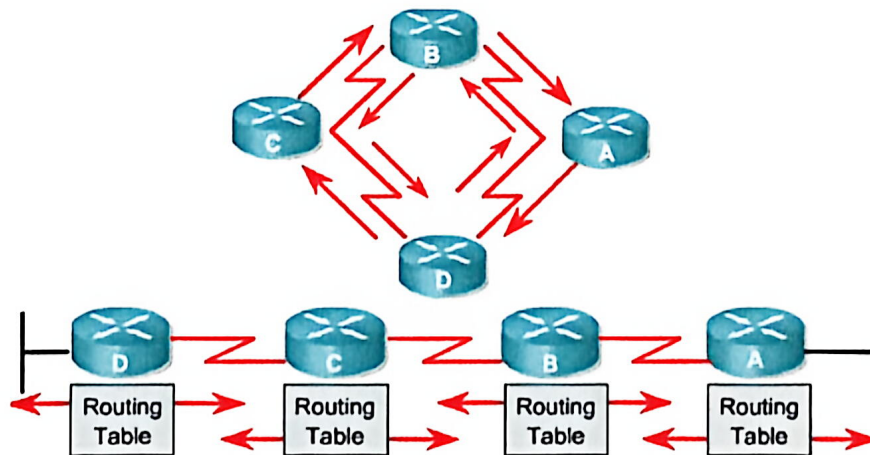
ISIS merupakan salah satu routing protocol yang diciptakan untuk OSI mode, menggunakan metode link state sebagai metode pengumpulan rutanya , ISIS juga akan melakukan pengumpulan informasi dan status dari semua link yang ada dalam jaringannya. Kemudian informasi tersebut dibentuk menjadi sebuah informasi rute yang dapat digunakan untuk meneruskan data. Selain ISIS , juga ada routing protocol yang mempergunakan metode link state yaitu OSPF, kedua routing protocol ini selain sama-sama menggunakan metode link state, juga menggunakan algoritma *Shortest Path First* (Algoritma Djikstra) dalam melakukan perhitungannya, kedua menggunakan konsep hierarki, system area dan banyak lagi[6].

Perbedaan Link State dan Distance Vector. Kedua metode pengumpul rute ini memiliki perbedaan yang sangat menonjol dalam hal algoritma yang digunakannya dalam proses pengumpulan rutanya.



Gambar 2.6. Klasifikasi *Routing Protocol*

Distance Vector melakukan pengumpulan rute dengan cara melakukan penghitungan jumlah perangkat router yang terbentang dari titik asal menuju ke tujuannya. Artinya setiap kali sebuah informasi melewati sebuah perangkat router, maka algoritma Distance Vector akan menghitungnya sebagai satu titik perjalanan, atau yang biasa disebut dengan istilah Hop.



Gambar 2.7. Konsep *Distance Vector*

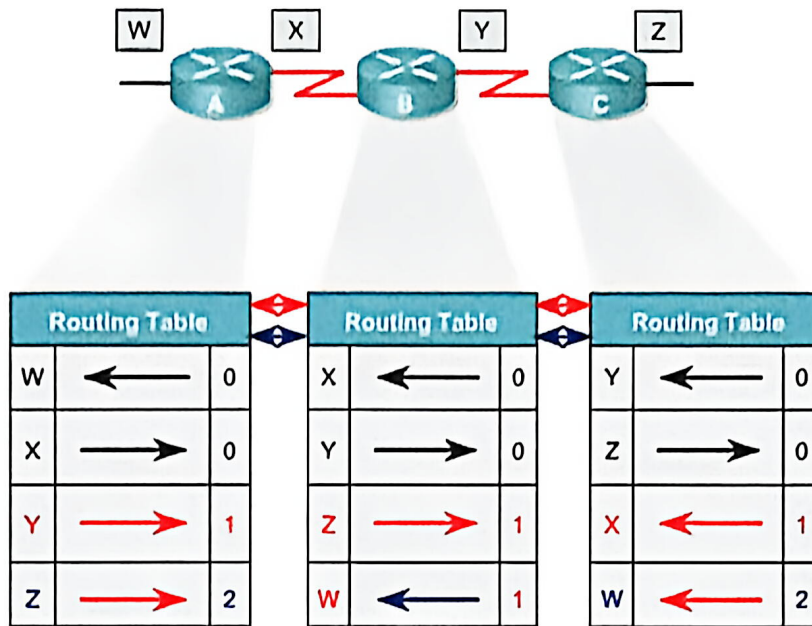
Pada Gambar 2.8 dapat dilihat bahwa Router B menerima informasi dari Router A. Router B menambahkan nomor *distance vector*, seperti jumlah *hop*. Jumlah ini menambahkan *distance vector*. Router B melewati *table routing* baru ini ke router-router tetangganya yang lain, yaitu Router C. Proses ini akan terus berlangsung untuk semua router.

Semakin banyak hop yang dilewati artinya semakin banyak perangkat yang dilewati dan semakin jauh jarak yang akan di tempuh oleh sebuah data. Dengan jarak yang semakin jauh tentu data dan informasi yang ingin di sampaikan akan memakan waktu perjalanan yang lama, dengan demikian informasi tidak dapat disampaikan dengan segera. Sebuah rute yang melewati banyak hop akan diartikan sebagai rute yang kurang baik.

Setiap routing protocol yang menggunakan metode Distance Vector pasti melakukan hal yang sama dalam proses pengumpulan rutenya. Mengumpulkan rute-rute yang ada, menghitung jumlah hop yang terdapat dalam rute tersebut, dan memilihnya salah satu yang terbaik yang kemudian dimasukkan kedalam routing table. Biasanya routing protocol jenis ini memiliki keterbatasan dalam jumlah hop nya. Artinya, routing protocol ini hanya mampu mencari rute dan meneruskan informasi meneruskan data yang akan dikirim ke tujuannya.

Distance Vector biasanya digunakan dalam jaringan komunikasi data dalam skala yang kecil hingga menengah, di mana jumlah perangkat jaringan yang saling dihubungkan tidak bisa terlalu banyak karena keterbatasan jumlah hop ini. Selain itu metode Distance Vector memiliki kelemahan yaitu mengirimkan semua routing table yang dimilikinya dalam satuan waktu tertentu, sehingga jika jaringan yang dilayaninya besar, tentu tabelnya akan semakin besar. Hal ini bisa mengakibatkan bandwidth jalur komunikasi dipenuhi oleh lalulintas routing table saja.

Setiap router menggunakan *distance vector* pertama kali mengidentifikasi router-router tetangganya. Interface terhubung langsung ke router tetangganya. Interface yang terhubung langsung ke router tetangganya mempunyai *distance* 0. Router menerapkan *distance vector* dapat menentukan jalur terbaik untuk menuju ke jaringan tujuan berdasarkan informasi yang diterima dari tetangganya. Router A mempelajari jaringan lain berdasarkan informasi yang diterima dari router B. Masing-masing router lain menambahkan dalam *table routing*-nya yang mempunyai akumulasi *distance vector* untuk melihat sejauh mana jaringan yang akan dituju, seperti yang ditunjukkan pada Gambar 2.9



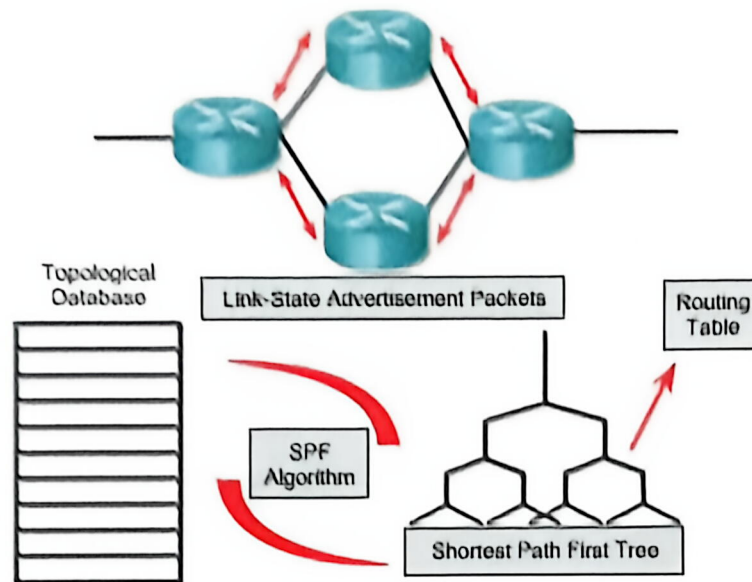
Gambar 2.8 Jaringan *Distance Vector Discovery*

Metode Link State bekerja dengan cara yang berbeda. Walau proses pengumpulan rutanya lebih rumit dan berat daripada metode Distance Vector, namun metode ini lebih reliabel, lebih skalabel dalam melayani jaringan besar, lebih terstruktur dan juga lebih menghemat bandwidth.

Metode Link State melakukan tracking atau penyelidikan terhadap semua koneksi yang ada dalam jaringan. Status dari koneksi-koneksi tersebut, jenis dan tipe koneksi, bahkan kecepatan dari koneksi tersebut semuanya dikumpulkan menjadi sebuah informasi. Kumpulan informasi tersebut kemudian di olah untuk menghasilkan daftar-daftar rute yang ada ke suatu tujuan. Kemudian melalui perhitungan itu, semua rute yang ada dipilih salah satu yang terbaik berdasarkan semua parameter yang dikumpulkannya. Hasilnya adalah sebuah rute terbaik yang langsung dapat digunakan oleh data untuk mencapai ke suatu tujuan.

Routing protokol metode Link State akan mengetahui ketika sebuah koneksi di dalam jaringan Anda putus atau mati. Ketika ada yang putus, proses perhitungan metode ini akan segera melepaskan rute-rute yang berhubungan dengan koneksi tersebut, sehingga tidak ada data yang nyasar dan terjebak di jalur putus tersebut. Metode Link State mungkin akan memilih sebuah rute yang didalamnya terdapat banyak hop namun lebih cepat koneksinya. Fitur-fitur yang dimiliki oleh *routing link-state* adalah :

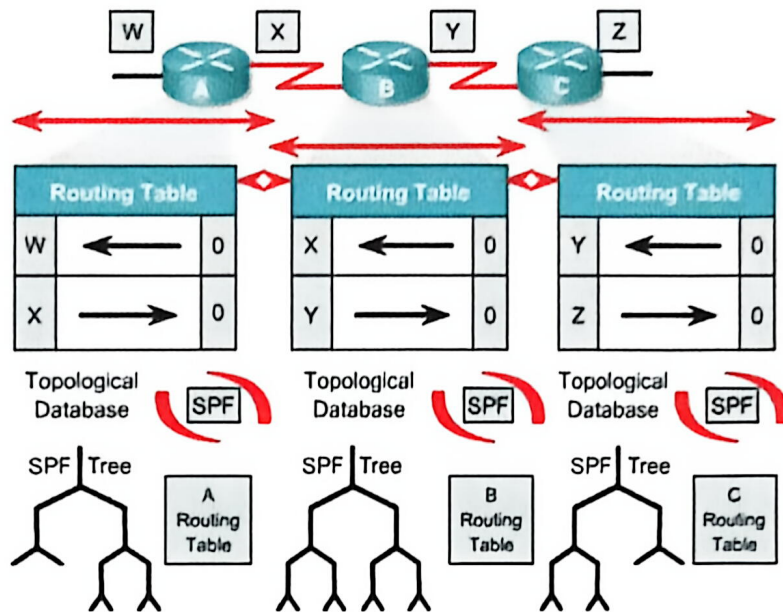
- *Link-State Advertisement (LSA)* : paket kecil dari informasi *routing* yang dikirim antar router
- *Topological database* : kumpulan informasi yang dari LSA-LSA
- *SPF algorithm* : hasil perhitungan pada database sebagai hasil dari pohon SPF
- *Routing Table* : daftar rute dan interface



Gambar 2.9 Konsep *Link State*

Ketika router melakukan pertukaran LSA, dimulai dengan jaringan yang terhubung langsung tentang informasi yang mereka miliki. Masing-masing router membangun database topologi yang berisi pertukaran informasi LSA.

Algoritma SPF menghitung jaringan yang dapat dicapai. Router membangun *logical* topologi sebagai pohon (*tree*), dengan router sebagai *root*. Topologi ini berisi semua rute-rute yang mungkin untuk mencapai jaringan dalam protokol *link-state internetwork*. Router kemudian menggunakan SPF untuk memperpendek rute. Daftar rute-rute terbaik dari interface ke jaringan yang dituju dalam *table routing*. *Link state* juga memperbaiki database topologi yang lain dari elemen-elemen topologi dan status secara detail.



Gambar 2.10. Jaringan *Link-State Discovery*

Pada Gambar 2.11 menunjukkan bahwa router pertama yang mempelajari perubahan topologi *link-state* melewati informasi sehingga semua router dapat menggunakannya untuk proses *update*. Informasi *routing* akan dikirim ke semua router dalam *internetwork*. Untuk mencapai keadaan konvergen, setiap router mempelajari router-router tetangganya. Termasuk nama dari router-router tetangganya, status interface dan *cost* dari *link* ke tetangganya. Router membentuk paket LSA yang mendaftarkan informasi ini dari tetangga-tetangga baru, perubahan *cost* *link* dan *link-link* yang tidak lagi *valid*. Paket LSA ini kemudian dikirim keluar sehingga semua router-router lain menerima itu.

2.3.1 Sistem Pengalamatan *Routing Protocol IS-IS*

Seperti yang sudah di bahas sebelumnya, IS-IS merupakan *routing protocol* yang diciptakan oleh OSI. OSI menggunakan *connectionless network protocol* (CLNP) *address*, dan ketika CLNP *address* digunakan dirouter maka disebut dengan *Network Service Access Point* (NSAP) dan NSAP ini yang digunakan dalam sistem pengalamatan IS-IS. NAP ini memiliki panjang yang bervariasi mulai dari 0 oktet sampai dengan 20 oktet.

2.3.2 Struktur Hirarki Pengalamatan

Untuk membedakan router mana yang merupakan router *backbone* area dan yang mana router *non-backbone* area. IS-IS menggunakan konsep tiga router. Tipe router dalam IS-IS dibagi menjadi :

- Router Level-1

Router tipe ini hanya bisa berkomunikasi dengan router dalam area yang sama

- Router Level-2

Router ini dapat berkomunikasi dengan router dalam area yang berbeda, router ini termasuk dalam *backbone* area router

- Router Level 1-2

Router dengan tipe ini seolah-olah memiliki dua router dalam satu perangkat, memiliki kemampuan level-1 dan kemampuan level-2

2.4. MPLS-TE LDP over RSVP

LDP adalah protokol yang paling sederhana dari protokol signaling di MPLS. Hanya dengan mengaktifkan LDP pada node / interface, label secara otomatis dapat diteruskan untuk setiap rute dalam tabel routing untuk semua neighbor LDP. LDP adalah protocol pembuat LSP yang sederhana. Full mesh LSP secara otomatis dibuat sehingga semua node saling berhubungan dengan LDP[8].

RSVP-TE menciptakan LSP ketika parameter untuk LSP dikonfigurasi secara manual. Jika parameter dapat dipenuhi berdasarkan informasi dalam database rekayasa lalu lintas, RSVP-TE signaling digunakan untuk membuat sambungan (LSP). Parameter ini meliputi tujuan dan mungkin termasuk pilihan seperti: kebutuhan bandwidth, atribut link, rute eksplisit, backup dan cepat reroute dan lain-lain.

Setup LSP menggunakan RSVP-TE dapat menjamin *bandwidth* yang diminta. Perlu diingat bahwa ini adalah di LSP-level, sehingga harus diperhatikan dalam mengelola lalu lintas dan layanan yang menggunakan LSP. Kegagalan dalam jaringan pasti ada, LSP yang dikonfigurasi untuk mengubah rute (FRR) dengan cepat dapat melewati kegagalan dalam waktu 50 ms atau kurang.

Hal yang perlu diperhatikan pada protokol RSVP-TE adalah bahwa setiap LSP harus ditetapkan secara manual. Bayangkan jika jaringan dengan router 20 PE yang harus terhubung dengan jala penuh LSP. Hal ini memerlukan konfigurasi 19 LSP pada masing-masing dari 20 node atau dikenal sebagai masalah n -square. Dan apa yang terjadi ketika salah satu simpul dihapus atau ditambahkan ke jaringan? Hal ini memerlukan konfigurasi yang harus dilakukan pada semua node. Ingat LDP secara otomatis membuat full mesh ini hanya dengan menghidupkan protokol.

2.4.1 Aplikasi LDP dan RSVP

Penyedia layanan seperti jaringan operator dapat menggunakan LDP LSP primer untuk *best-effort* trafik dan bagi pelanggan yang menggunakan “*bare bones*“, jenis murah layanan. Operator mungkin juga menggunakan RSVP-TE LSP untuk para pelanggan yang bersedia membayar ekstra untuk jaminan *bandwidth* dan ultra kehandalan. Dari perspektif konfigurasi, kebanyakan vendor mendukung kedua protokol secara bersamaan, sehingga benar-benar hanya masalah implementasi. Pertanyaannya dimana menggunakan setiap protokol, jika mengaktifkan keduanya dimana-mana, ada beberapa kesulitan :

- a) Duplikat *full-mesh* LSP – banyak *overhead* sinyal

Mengidentifikasi lalu lintas menggunakan yang LSP atau harus statis memetakan arus lalu lintas ke LSP

- b) *Reconvergence* dari LDP LSP dalam hal link / kegagalan node

Pilihan lain adalah dengan mengaktifkan LDP dimana-mana, dan membangun jalur RSVP-TE yang diperlukan bagi para pelanggan yang bersedia untuk membayar premi. Maka pertanyaannya menjadi apakah setiap pelanggan premium mendapatkan set mereka sendiri dalam 1 LSP atau apakah mereka dibagi antara pelanggan? Jika setiap pelanggan mendapat b LSP TE, maka penyedia mungkin harus menjalankan beberapa LSP antara sama 2 *endpoint*, membutuhkan lebih banyak administrasi dan manajemen. Ini masalah implementasi, sehingga setiap penyedia layanan harus memutuskan bagaimana untuk *roll-out* penawaran.

2.5.1 Parameter QoS

a) *Rate*

Rasio jumlah bits yang dipindahkan/ditransmisikan antar dua perangkat dalam satuan waktu tertentu, umumnya dalam detik. *Bit Rate* sama dengan istilah lain *data rate*, *data transfer rate* dan *bit time*.

b) *Delay*

Delay adalah waktu yang dibutuhkan data untuk menempuh jarak dari asal ke tujuan. *Delay* dapat dipengaruhi oleh jarak, media fisik, kongesti atau juga waktu proses yang lama.

Tabel 2.2. One-Way Delay/Latency [9]

Kategori <i>Latency</i>	Besar <i>Delay</i>
<i>Excellent</i>	<150 ms
<i>Good</i>	150 s/d 300 ms
<i>Poor</i>	300 s/d 450 ms
<i>Unacceptable</i>	>450 ms

c) *Packet Loss* atau *Error*

Packet Loss merupakan suatu parameter yang menggambarkan suatu kondisi yang menunjukkan jumlah total paket yang hilang. Salah satu penyebab *packet loss* adalah antrian yang melebihi kapasitas *buffer* pada setiap *node*. Beberapa penyebab terjadinya *packet loss* yaitu :

- *Congestion*, disebabkan terjadinya antrian yang berlebihan dalam jaringan
- *Node* yang bekerja melebihi kapasitas *buffer*
- *Memory* yang terbatas pada *node*
- *Policing* atau kontrol terhadap jaringan untuk memastikan bahwa jumlah trafik yang mengalir sesuai dengan besarnya *bandwidth*. Jika besarnya trafik yang mengalir didalam jaringan melebihi dari kapasitas *bandwidth* yang ada maka *policing control* akan membuah kelebihan trafik yang ada

Tabel 2.3. Packet Loss [9]

Kategori Degradasi	Packet Loss
Sangat Bagus	0
Bagus	3%
Sedang	15%
Jelek	25%

d) Jitter

Jitter didefinisikan sebagai variasi dari *delay* atau selisih *delay* pertama dengan *delay* setelahnya. *Delay* yang diakibatkan oleh panjang antrian dalam suatu pengolahan data dan *reassemble* paket-paket data di akhir pengiriman akibat kegagalan sebelumnya. *Jitter* merupakan masalah yang khas pada *connectionless* atau *packet switch network*.

Secara umum *jitter* merupakan masalah *slow speed links*. Karena itu diharapkan peningkatan QoS dengan mekanisme *priority buffer*, *bandwidth reservation* dan *high speed connections* dapat mereduksi masalah *jitter* di masa depan. Cisco menetapkan bahwa *jitter* untuk komunikasi *realtime* seperti video interaktif dan *voice* tidak boleh melebihi 30 ms. Semakin besar nilai *Jitter* akan mengakibatkan nilai QoS akan semakin turun. Untuk mendapatkan nilai QoS jaringan yang baik, nilai *jitter* harus dijaga sekecil mungkin.

Tabel 2.4. Jitter [9]

Kategori Degradasi	Peak Jitter
Sangat Bagus	0
Bagus	0 s/d 75 ms
Sedang	76 s/d 125 ms
Jelek	125 s/d 225 ms

e) Throughput

Throughput adalah kecepatan (*rate*) transfer data efektif yang diukur dalam bps. *Throughput* merupakan jumlah bit yang sukses dikirim

dari suatu terminal tertentu didalam sebuah jaringan, atau dari suatu titik ke titik jaringan yang lain dibandingkan dengan total waktu pengiriman. *Throughput* maksimal dari suatu titik atau jaringan komunikasi menunjukkan kapastitasnya. Berikut adalah perhitungan rumus dalam mencari nilai *throughput* :

$$Throughput = \frac{Paket\ diterima}{Time\ between\ first\ and\ last\ packet} \quad (2.1)$$

2.5.2 Pentingnya QoS

Ada beberapa alasan mengapa diperlukan QoS, yaitu :

- Untuk memberikan prioritas untuk aplikasi-aplikasi yang kritis pada jaringan
- Untuk memaksimalkan penggunaan investasi jaringan yang sudah ada
- Untuk meningkatkan performansi untuk aplikasi-aplikasi yang sensitif terhadap delay, seperti *Voice* dan *Video*
- Untuk merespon terhadap adanya perubahan-perubahan pada aliran trafik di jaringan

2.6. Key Quality Indicators (KQI)

Key Quality Indicators (KQI) dalam hal ini *Page Download Throughput* adalah parameter yang dipakai untuk melihat seberapa besar kecepatan internet yang digunakan oleh *subscriber* di daerah tertentu tergantung juga oleh *coverage* areanya. Tentang rata-rata kecepatan internet pada jaringan 2G/3G/4G pada suatu daerah dan tergantung juga ada atau tidaknya problem (alarm) pada site tersebut. Parameter yang dapat dilihat pada KQI ini adalah *Throughput* dan *Delay*.

Untuk menghitung besarnya degradasi/penurunan *throughput* digunakan rumus sebagai berikut :

$$\Delta \text{ Degradasi } Throughput (\%) = \frac{A-B}{A} \times 100\% \quad (2.2)$$

Keterangan :

- A = 1 Day Ago (Kbps)
- B = Current Period (Kbps)

Sedangkan untuk menghitung besarnya degradasi/kenaikan *delay* digunakan rumus sebagai berikut :

$$\Delta \text{ Degradasi Delay (\%)} = \frac{B-A}{A} \times 100\% \quad (2.3)$$

Keterangan :

- A = 1 Day Ago (Kbps)
- B = Current Period (Kbps)