

## BAB II

### LANDASAN TEORI

#### 2.1 Tinjauan Pustaka

Pelaksanaan Penelitian ini meninjau beberapa hasil penelitian sebelumnya sebagai tinjauan pustaka. Penelitian-penelitian tersebut dijadikan sebagai referensi dalam menentukan arah pelaksanaan penelitian ini. Hasil penelitian yang dijadikan tinjauan pustaka yaitu:

Penelitian yang berjudul “*Equal Cost Multipath Routing in IP*”. Penelitian yang dilakukan oleh Neeraj Verma dkk. (2014), bertujuan untuk mengetahui load balancing dan *fast protection fitur* baru yang diimplementasikan menggunakan ECMP dan *hardware* dari Tellabs *router*. Poin kunci dari penelitian ini adalah Kinerja algoritma *load balancing*.

Hasil penelitian ini yaitu dalam *load balancing*, *traffic* distribusi antar *link* yang berbeda dan kecepatan *bandwidth* lebih tinggi dibandingkan dengan menggunakan satu jalur saja.

Penelitian selanjutnya yang berjudul “Analisa perbandingan metode *load balancing peer connection classifier* (PCC) dengan NTH pada *router* mikrotik”.

Penelitian yang dilakukan oleh Agung Rahmawan Saputra. 2013, bertujuan untuk mengoptimalkan dua koneksi internet agar dapat digunakan secara efektif dengan pemanfaatan dua jalur *gateway* yang berbeda, agar kebutuhan internet tercukupi sesuai kebutuhan. Metode *load balance* yang digunakan yaitu NTH dan PCC menggunakan modem CDMA dan GSM.

Hasil penelitian ini menunjukkan bahwa NTH *load blancing* dapat membagi *traffic* jaringan secara adil dan meminimalisir terjadinya *overload* pada salah satu koneksi ISP, akan tetapi sering terjadi diskoneksi untuk aplikasi *realtime* dikarenakan perpindahan *gateway* dapat diminimalkan dengan dibuat peraturan tambahan. Sedangkan PCC *load balancing* mampu menspesifikasikan *gateway* untuk tiap paket yang masih berhubungan dengan data yang sebelumnya dilewatkan pada salah satu *gateway*, tapi sering terjadi *overload* pada salah satu *gateway* yang disebabkan pengaksesan situs yang sama. Namun, PCC dapat lebih mengoptimalkan penggunaan dua koneksi internet dibandingkan dengan Nth.

Penelitian selanjutnya yang berjudul “Implementasi dan analisa metode *load balancing* pada layanan Telkom speedy”. Penelitian yang ditulis oleh Andri Suwignyo dkk. (2011), bertujuan untuk memberikan solusi ekonomis dan memaksimalkan penggunaan *resource bandwidth* yang dimiliki. Dalam penelitian ini, implementasi metode *load balancing* dilakukan dengan metode ECMP, Nth, dan PCC pada layanan Telkom Speedy. Melakukan analisa metode *load balancing* yang digunakan terhadap utilisasi masing-masing jalur speedy, dan dilakukan pula analisa terhadap beban trafik terhadap skenario *limit bandwidth* yang digunakan.

Hasil dari penelitian ini yaitu metode Nth memberikan nilai rata-rata utilisasi semua jalur paling tinggi pada ketiga jalur speedy, pada kondisi tanpa *bandwidth*, dengan nilai prosentase sebesar 93,27%, dibandingkan dengan metode PCC dan ECMP dengan nilai prosentase berturut-turut sebesar 9,81% dan 83,57%.

Besar *limit bandwidth* berpengaruh terhadap persebaran beban trafik telah sesuai dengan kemampuan tiap jalur, dimana pada jalur 2 mendapatkan 50,00% dari total

trafik, dan pada jalur 1 dan 3 mendapatkan masing-masing 25,00% dari total trafik.

Mekanisme *load balancing* dengan menggunakan metode *Nth* dapat diimplementasikan pada *Atmosphere Network* Bandung untuk memaksimalkan tiga jalur speedy yang tersedia, sehingga utilisasi tiap jalur bisa lebih maksimal dibandingkan menggunakan metode ECMP dan PCC.

Kesimpulan dari ketiga penelitian di atas yaitu *traffic* distribusi antar *link* yang berbeda dan kecepatan *bandwidth* lebih tinggi dibandingkan dengan menggunakan satu jalur saja. *PCC load balancing* mampu menspesifikasikan *gateway* untuk tiap paket yang masih berhubungan dengan data yang sebelumnya dilewatkan pada salah satu *gateway*, tapi sering terjadi *overload* pada salah satu *gateway* yang disebabkan pengaksesan situs yang sama.

Kesimpulan dari peninjauan ketiga penelitian di atas adalah dengan memiliki beberapa *link* atau jalur, trafik dapat berjalan optimal, memperkecil waktu tanggap dan menghindari *overload*, *link* dapat saling mem-*backup*. Beban setiap *link* dibagi ke beberapa jalur menggunakan *load balancing*, agar tidak ada *link* yang mendapatkan beban lebih besar dari *link* yang lain. Dengan pembagian beban tersebut, maka diharapkan akan tercapai keseimbangan (*balance*) penggunaan *link-link* tersebut. Serta besar *limit bandwidth* berpengaruh terhadap persebaran beban trafik telah sesuai dengan kemampuan tiap jalur

## 2.2 Landasan Teori

### 2.2.1 Router

Menurut (Siddik., Dkk, 2015), *router* adalah perangkat keras yang memfasilitasi transmisi paket data melalui jaringan komputer. *Router* merupakan perangkat jaringan yang bekerja pada OSI layer 3. *Router* berfungsi sebagai penghubung antara dua atau lebih jaringan untuk meneruskan data dari satu jaringan ke jaringan lainnya. *Router* berbeda dengan *switch*. Sebagai ilustrasi perbedaan fungsi *router* dan *switch* adalah *switch* merupakan sebuah jalan, dan *router* merupakan penghubung antar jalan. Masing-masing rumah berada pada jalan yang memiliki alamat dalam suatu urutan tertentu. Beberapa routerboard biasanya sudah terinstalasi MikroTik Os.



**Gambar 2.1** Routerboard (Sumber: Jurnal SAINTIKOM Vol. 14 No. 1, Januari 2015)

Menurut (Wirawan, 2011), *router* adalah salah satu komponen pada jaringan komputer yang mampu melewatkan data melalui sebuah jaringan atau internet menuju sasarannya, melalui sebuah proses yang dikenal sebagai *routing*. Proses *routing* dapat dilakukan dengan memasukkan informasi suatu alamat jaringan secara manual ke dalam *table routing* ataupun dengan bantuan *protocol routing*.

Sebuah *router* mampu mengirimkan data/informasi dari satu jaringan ke jaringan lain yang berbeda, *router* hampir sama dengan *bridge* hanya mampu menghubungkan jaringan yang sama. Dan dalam pengembangan perangkat *router* dewasa ini sudah mulai mencapai bahkan melampaui batas tuntutan teknologi yang diharapkan. *Router* akan mencari jalur terbaik untuk mengirimkan sebuah pesan yang berdasarkan atas alamat tujuan dan alamat asal. *Router* mengetahui alamat secara keseluruhan dari masing-masing komputer di lingkungan jaringan lokalnya, dan *router* lainnya.

### 2.2.2 Gateway

Menurut (Wirawan, 2011), *gateway* dapat diartikan sebagai pintu gerbang sebagai keluar masuknya paket data dari *local network* menuju *router network*. Tujuannya agar *client* pada *local network* dapat berkomunikasi dengan *network* lainnya atau internet. *Router* dapat di *setting* menjadi *gateway* dimana ia menjadi penghubung antara jaringan *local* dengan jaringan luar.



### 2.2.3 Routing Protokol

Menurut (wirawan, 2011) *router* mempunyai fungsi untuk menghubungkan 2 atau lebih *network* yang berbeda. *Router* mempunyai *routing table* yang digunakan sebagai dasar dalam pencarian jalur menuju *network* yang dituju oleh paket. Bila terdapat lebih dari satu jalur menuju *network* tujuan, maka *router* akan mencari jalur yang terbaik menurut aturan “*best path*” yang dimilikinya, jalur-jalur tersebut dinilai sama baik.

Isi dari *routing table* bisa didapat melalui berbagai cara:

- a. *Static Route*, ini adalah jenis *route* yang diinput secara manual ke dalam table routing. *Route* jenis ini hanya cocok digunakan bila skala *network* tidak terlalu besar atau ketika diperlukan *route* khusus ke sebuah *network*, karena sering bertambahnya jumlah *segment/network*, maka jumlah *static route* dan juga *gateway* tempat paket dikeluarkan akan meningkat.
- b. *Default route* ini adalah *route* yang digunakan ketika *router* tidak dapat menemukan jalur lain menuju *network* tujuan yang lebih spesifik, *default route* sebenarnya adalah *static route* dengan format 0.0.0.0/0.
- c. *Dynamic route*, *route* jenis ini akan di input ke dalam *routing table* dan bantuan dari *routing protocol*. Jenis *route* ini akan cukup dideklarasikan saja (menggunakan *comment “network”*), semua *routing protocol* pada *router* akan menginput ke dalam *routing*

*table* dan mencari “*gateway*” atau jalur keluar bagi paket berdasarkan algoritma yang dijalankan.

#### 2.2.4 *Load Balancing*

*Load Balance* dalam jaringan komputer adalah teknik untuk membagi beban (*load*) ke dalam beberapa jalur atau *link*. Ini dilakukan jika untuk menuju suatu *network* terdapat beberapa jalur (*link*). Tujuan dari *load balance* ini agar tidak ada *link* yang mendapatkan beban lebih besar dari *link* yang lain. diharapkan dengan membagi beban ke dalam beberapa *link* tersebut, maka akan tercapai keseimbangan (*balance*) penggunaan *link-link* tersebut (Towidjojo, 2013).

*Load balancing* adalah teknik untuk mendistribusikan beban trafik pada dua atau lebih jalur koneksi secara seimbang, agar trafik dapat berjalan optimal, memaksimalkan *throughput*, memperkecil waktu tanggap dan menghindari *overload* pada salah satu jalur koneksi. (Dewobroto, 2009).

Menurut (Hafizh, 2011), secara umum, *load balancing* dapat diartikan sebagai suatu teknik untuk mendistribusikan beban kerja secara merata pada dua atau lebih komputer, *network links*, CPU, *hard drive* atau sumber daya lainnya, untuk mendapatkan pemanfaatan sumber daya yang optimal, memaksimalkan *throughput*, meminimalkan waktu respon dan menghindari *overload*. Menggunakan beberapa komponen dengan *load balancing* dapat meningkatkan kehandalan melalui redundansi. Layanan *load balancing*

biasanya disediakan oleh program khusus atau perangkat keras (seperti *multilayer switch* atau *DNS sever*).

*Load balancing* adalah suatu proses dan teknologi yang data mendistribusikan *trafik* ke beberapa *server* dengan menggunakan komputer/perangkat jaringan. Proses ini mampu mengurangi beban kerja setiap *server*, serta memunginn *server* untuk menggunakan *bandwidth* yang tersedia secara lebih efektif. Keefektifan kinerja *server* dapat dilihat dari dua parameter berikut:

a) *Response Time*

*Response Time* adalah waktu yang dibutuhkan oleh *server* untuk memroses sebuah paket data yang datang dan menghasilkan sebuah paket data balasan.

b) *Throughput*

*Throughput* adalah *bandwidth* aktual yang terukur pada ukuran waktu tertentu. *Throughput* dapat merepresentasikan kemampuan sebuah *server* untuk menyalurkan data dalam jaringan dalam satuan detik, oleh karena itu *throughput* menggambarkan *bandwidth* yang sebenarnya (aktual) pada suatu waktu tertentu.

Pada *load balancing* juga terdapat istilah *high availability*, yaitu merupakan metode dimana *load balancer* dijaga keberadaannya dan dapat dimanfaatkan atau mengambil alih proses yang dilakukan oleh *load balancer* lain bilamana dibutuhkan.



Dalam jaringan komputer, *load balancing* lebih mengarah kepada pengkombinsian beberapa antarmuka *Ethernet* ke dalam satu jalur sehingga dapat diutilasi secara bersamaan dengan menghasilkan koneksi yang lebih cepat.

Untuk dapat mengimplementasikan sistem ini diperlukan perangkat tambahan berupa *router* MikroTik.

Dengan konsep yang sederhana, sebuah *load balancer* diletakkan di antara *client* dan *server*. *Load balancer* akan menampung yang datang dan membaginya ke dalam *request-request* individual lalu menentukan *server* mana yang menerima *request* tersebut.

Menurut (Sagala, 2010), Beberapa kemungkinan dari penerapan *load balancing* antara lain:

1. *Scalability* : Ketika beban sistem meningkat, kita dapat melakukan perubahan terhadap sistem agar dapat mengatasi beban sesuai dengan kebutuhan.
2. *High Availability* : *load balancer* secara terus menerus melakukan pemantauan terhadap *server*. Jika terdapat *server* yang mati, maka *load balancer* akan menghentikan *request* ke *server* tersebut dan mengalihkannya ke *server* yang lain.
3. *Manageability* : mudah ditata meskipun secara fisik sistem sangat besar.

4. *Security* : Untuk semua trafik yang melewati *load balancer*, aturan keamanan dapat diimplementasikan dengan mudah. Dengan *private network* digunakan untuk *server*, alamat IP nya tidak akan diakses secara langsung dari luar sistem.

Saat sebuah *router* yang mempunyai dua koneksi ke internet (sama atau berbeda ISP-nya), *default gateway* di *router* tetap hanya bisa satu, ditambah pun yang bekerja tetap hanya satu. Jadi misal *router* NAT terhubung ke ISP A melalui *interface* A dan *gateway* A dan ke ISP B melalui *interface* B dan *gateway* B, dan *default gateway* ke ISP A, maka trafik *downlink* hanya akan datang dari ISP A saja. Begitu juga sebaliknya jika dipasang *default gateway* ke ISP B. Penerapan teknik *load balancing* dapat menyelesaikan permasalahan tersebut dengan menggabungkan trafik *downlink* ISP A dan ISP B sehingga dapat diutilisasi secara bersamaan.

Prinsip dari *load balancing* adalah sebagai berikut:

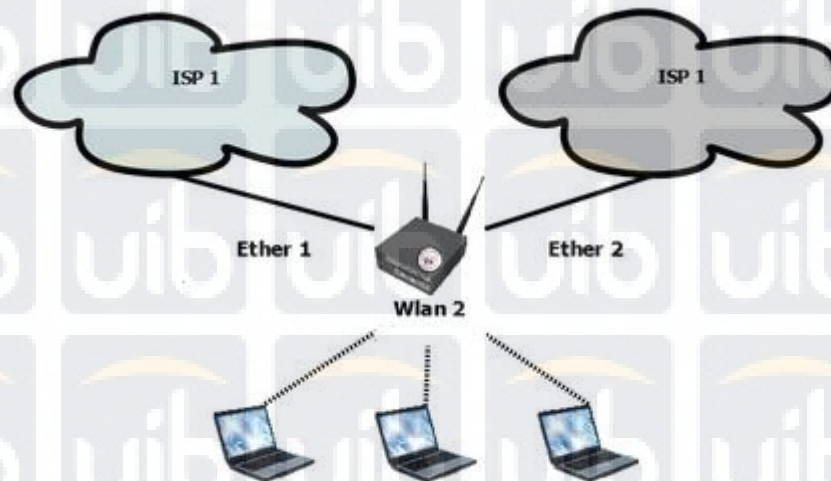
1. Lalu lintas didistribusikan berdasarkan probabilitas.
2. Harus tau seberapa besar tiap *link*, dan didistribusikan sesuai lalulintas.
3. Berdasarkan kecepatan pada keluaran dan masukan pada *router*, *load balancing* dapat diilustrasikan sebagai berikut:

$$512 + 256 \neq 768$$

$$512 + 256 = 256 + 512$$

$$512 + 256 = 128 + 128 + 128 + 128 + 128 + 128$$

4. Jika ada dua *gateway*, misal A dan B, A memiliki *bandwidth* sebesar 1 Mbps dan B memiliki *bandwidth* sebesar 2 Mbps. Maka lalulintas akan dibagi ke dalam 3 aliran, dan mengirim 1 aliran ke A dan 2 aliran ke B.



**Gambar 2.2** Ilustrasi *Load balancing* (Sumber : [www.mikrotik.co.id](http://www.mikrotik.co.id))

Menurut (Utomo Dwi, 2011) ada berbagai metode *load balancing*, antara lain *static route* dengan *address list*, *Equal Cost Multi path (ECMP)*, *Nth* dan *Per Connection Classifier (PCC)*. Setiap metode *load balancing* tersebut memiliki kekurangan maupun kelebihan tersendiri. Namun lebih dari hal itu, yang paling terpenting dalam menentukan metode *load balancing* apa yang akan digunakan adalah harus terlebih dahulu mengerti karakteristik dari jaringan yang akan diimplementasikan. Berikut ini adalah sedikit pengertian dari masing-masing metode *load balancing* dan disertakan pula kekurangan maupun kelebihanannya.

#### 2.2.4.1 *Static Route dengan Address list*

*Static route dengan address list* adalah metode *load balancing* yang mengelompokkan suatu *range IP address* untuk diatur agar dapat melewati salah satu *gateway* dengan menggunakan *static routing*.

Metode ini sering digunakan pada warnet yang membedakan PC untuk *browsing* dan PC untuk *game online*. Mikrotik akan menentukan jalur *gateway* yang dipakai dengan membedakan *src-address* pada paket data.

#### 2.2.4.2 *Equal Cost Multi Path (ECMP)*

*Equal Cost Multi Path (ECMP)* adalah pemilihan jalur keluar secara bergantian pada *gateway*. Contoh jika ada dua *gateway* dia akan melewati kedua *gateway* tersebut dengan beban yang sama (*equal cost*) pada masing-masing *gateway*.

Nilai dari *equal cost* dapat pula didefinisikan secara asimetris atau tidak seimbang pada saat *routing*. Ini dikarenakan jika diantara kedua ISP memiliki kecepatan koneksi yang berbeda jauh. Contoh jika kedua *gateway* mempunyai kecepatan koneksi sebesar 1 Mbps dan 3 Mbps, maka pada saat konfigurasi *routing* akan menjadi “ip route add dst-address=0.0.0.0 gateway=10.10.0.1, 10.10.0.2, 10.10.0.2, 10.10.0.2 check –

gateway=ping” yang diartikan bahwa *gateway* pertama dan kedua berbanding terbalik 1:3.

#### 2.2.4.1 Nth

Nth bukanlah sebuah singkatan, melainkan Nth adalah sebuah integer (bilangan ke-N). Nth menggunakan algoritma *round robin* yang menentukan pembagian pemecahan *connection* yang akan di-*mangle* ke rute yang dibuat untuk *load balancing*.

Pada dasarnya koneksi yang masuk ke proses di *router* akan menjadi satu arus yang sama, walaupun mereka datang dari *interface* yang berbeda. Maka pada saat menerapkan metode Nth, tentunya akan memberikan batasan ke *router* untuk hanya memproses koneksi dari sumber tertentu saja. Ketika *router* telah membuat semacam antrian baru untuk batasan yang kita berikan di atas, baru proses Nth dimulai.

Di dalam Nth terdapat variabel yang harus dimengerti, yaitu:

1. *Every*: Angka *every* adalah jumlah kelompok yang ingin dihasilkan jadi bila administrator ingin membagi alur koneksi yang ada menjadi 2 kelompok yang nantinya akan di *load balance* ke 2 koneksi yang ada, maka angka *every*=2
2. *Packet*: Angka *packet* adalah jumlah koneksi yang akan ditandai atau di-*mangle*. Jika ingin membuat dua kelompok,



tentunya harus membuat 2 *mangle rules*. Pada *rules* tersebut, angka untuk *every* haruslah sama, namun untuk angka *packet* haruslah berubah. Untuk 2 kelompok, berarti angka *packet* untuk 2 *rules* tersebut adalah 1 dan 2.

3. *Counter*: *counter* atau disebut penghitung atau pancacah biner.

Mulai dari mikrotik versi 3.x nilai *counter* tidak didefinisikan langsung oleh administrator. Setiap *rules* memiliki *counter* sendiri. Ketika *rules* menerima paket, maka *counter* untuk aturan saat itu akan otomatis bertambah satu. Dan jika nilai *counter* sama dengan nilai “*every*”, maka paket akan dicocokkan dan *counter* akan diatur ke nilai awal.

#### 2.2.4.1 *Per Connection Classifier (PCC)*

*Per Connection Classifier (PCC)* merupakan metode yang menspesifikasikan suatu paket menuju *gateway* koneksi tertentu. PCC mengelompokkan trafik koneksi yang melalui atau keluar masuk router menjadi beberapa kelompok. Pengelompokkan ini bisa dibedakan berdasarkan *src-address*, *dst-address*, *src-port* dan atau *dst-port*. Mikrotik akan mengingat-ingat jalur *gateway* yang telah dilewati di awal trafik koneksi, sehingga pada paket-paket data selanjutnya yang masih berkaitan akan dilewatkan pada jalur *gateway* yang sama dengan paket data sebelumnya yang sudah dikirim.

**Table 2.1** Perbedaan masing-masing metode *load balancing*

Metode	Kelebihan	Kekurangan
<i>Static Route</i> dengan <i>Address List</i>	<ul style="list-style-type: none"> <li>• Dapat membuat topologi jaringan yang sederhana</li> <li>• Tidak ada diskoneksi pada <i>client</i> yang disebabkan perpindahan <i>gateway</i> karena <i>load balancing</i></li> </ul>	<ul style="list-style-type: none"> <li>• Dapat terjadi <i>overload</i> jika yang aktif hanya pada <i>client-client</i> pada salah satu <i>address list</i> saja.</li> </ul>
ECMP	<ul style="list-style-type: none"> <li>• Dapat membagi beban jaringan berdasarkan perbandingan kecepatan di antara 2 ISP</li> </ul>	<ul style="list-style-type: none"> <li>• Sering terjadinya diskoneksi yang disebabkan oleh <i>routing table</i> yang merestart secara otomatis setiap 10 menit</li> </ul>
Nth	<ul style="list-style-type: none"> <li>• Dapat membagi penyebaran paket data yang merata pada masing-masing <i>gateway</i></li> </ul>	<ul style="list-style-type: none"> <li>• Kemungkinan terjadi terputusnya koneksi yang disebabkan perpindahan <i>gateway</i> karena <i>load balancing</i></li> </ul>
PCC	<ul style="list-style-type: none"> <li>• Mampu menspesifikasi <i>gateway</i> untuk tiap paket data yang masih berhubungan dengan data yang sebelumnya sudah dilewatkan pada salah satu <i>gateway</i></li> </ul>	<ul style="list-style-type: none"> <li>• Dapat terjadinya <i>overload</i> pada salah satu <i>gateway</i> yang disebabkan pengaksesan situs yang sama</li> </ul>

### 2.2.6 Jaringan Komputer

Menurut (Sofana, 2011), Jaringan adalah himpunan interkoneksi sejumlah komputer *autonomous*. Kata “*autonomous*” mengandung pengertian bahwa komputer tersebut memiliki kendali atas dirinya sendiri. Bukan merupakan bagian komputer lain, seperti sistem terminal yang biasa digunakan pada komputer *mainframe*. Komputer juga tidak mengendalikan komputer lain yang dapat mengakibatkan komputer lain *restart*, *shutdown*, merusak *file*, dan sebagainya.

Dua buah komputer diartikan “*interkoneksi*” apabila keduanya bisa berbagi *resource* yang dimiliki, seperti saling bertukar data/informasi, berbagi *printer*, berbagi media penyimpanan (*hard disk*, *floppy disk*, *CD ROM*, *Flash Disk*, dan sebagainya).

Menurut (Irawan & Baraja, 2012), jaringan komputer adalah sekelompok komputer otonom yang saling berhubungan antara satu dengan yang lainnya menggunakan *protocol* komunikasi melalui media komunikasi sehingga dapat saling berbagi informasi, program-program, penggunaan bersama perangkat keras seperti *printer*, *hard disk*, dan sebagainya. Selain itu jaringan komputer bisa diartikan sebagai kumpulan sejumlah terminal komunikasi yang berada di berbagai lokasi yang terdiri dari lebih dari satu komputer yang saling berhubungan.

Jaringan komputer mampu bertindak sebagai medium komunikasi baik bagi para pegawai yang terpisah jauh. Dengan menggunakan jaringan, dua orang atau lebih yang tinggal berjauhan akan lebih mudah berkerjasama

menyusun laporan. Pada saat seorang pegawai membuat suatu perubahan di suatu dokumen *online*, pegawai lainnya dapat mengetahui perubahan tersebut dengan segera, tidak perlu lagi menunggu datangnya surat yang biasanya akan tiba beberapa hari kemudian. Percepatan seperti itu akan mengakibatkan kerjasama di antara komputer-komputer kerja yang terpisah berjauhan akan lebih mudah.

Di dalam suatu jaringan komputer terdapat beberapa tipe yaitu salah satu LAN (*Local Area Network*), yang digunakan untuk menghubungkan komputer yang berada di dalam suatu unit kecil, misalnya di dalam suatu gedung perkantoran atau kampus. Jarak antar komputer yang dihubungkannya bisa mencapai 5 sampai 10 KM dengan kecepatan 10 Mbps sampai 100 Mbps.

LAN menjadi populer karena memungkinkan banyak pengguna untuk memakai sumber daya secara bersama-sama. Contoh dari sumber daya yang dapat digunakan itu misalnya suatu *server*, *printer* dan sebagainya.

Menurut (Utomo dwi, 2011) Jaringan komputer adalah kumpulan dua atau lebih komputer yang saling berhubungan untuk melakukan komunikasi data. Komunikasi data yang bisa dilakukan melalui jaringan komputer dapat berupa data eks, gambar, video, dan suara. Untuk membangun sebuah jaringan komputer harus diperhatikan tentang situasi dan kondisi organisasi yang akan membangun jaringan tersebut, misalnya struktur bangunan, jangkauan, kecepatan akses, biaya operasional, dan sebagainya.

Dilihat dari sisi geografis, jaringan komputer dapat dibagi menjadi tiga bagian, yaitu:

a) *Local Area Network (LAN)*

LAN merupakan jaringan komputer dengan ruang lingkup terbatas, meliputi lokasi gedung, kampus, kantor, atau pabrik. Tipe ini banyak digunakan untuk perkantoran, bisnis, laboratorium, dan sebagainya dengan skala kecil seperti warnet, rental komputer, laboratorium komputer dan sebagainya.

Sebuah LAN dapat dibangun dengan minimal 2 (dua) komputer dengan spesifikasi (kapasitas) komputer rendah sekalipun. Adanya LAN akan menjadikan komputer terhubung dengan komputer lain, sehingga komputer tersebut seolah menjadi satu kesatuan dan bisa saling berinteraksi.

b) *Metropolitan Area Network (MAN)*

Metropolitan Area Network (MAN), adalah sebuah jaringan yang menggunakan teknologi yang sama dengan LAN, hanya saja ukurannya biasanya lebih luas daripada LAN. Jenis jaringan komputer ini adalah jaringan komputer yang memungkinkan jarak yang cukup jauh. Tipe ini digunakan untuk membangun jaringan komputer antar gedung, dalam satu kota, atau antar kota yang berbeda pada jangkauannya. Jaringan ini biasanya digunakan oleh perusahaan-perusahaan besar seperti perbankan, BUMN, perusahaan penjualan motor, dan lain-lain.



c) *Wide Area Network* (WAN)

Jaringan jenis ini merupakan jaringan terbesar karena mencakup radius antar Negara bahkan benua tanpa batasan geografis seperti jenis jaringan yang lain. WAN terdiri dari kumpulan LAN, MAN dan mesin-mesin yang bertujuan untuk menjalankan program aplikasi pemakai.

Menurut (Susilo dan purnama, 2012), berikut beberapa alasan dalam penggunaan jaringan komputer yaitu:

- a) Tidak terjadinya ketergantungan pada suatu merk komputer atau vendor tertentu sehingga memungkinkan berbagai merk komputer saling berhubungan.
- b) Pemakaian sumber daya secara bersama-sama. Ini dikarenakan keterbatasan sumber daya yang tersedia sehingga perlu dilakukan pembagian (*sharing*) dalam pemakaian sumber daya tersebut.
- c) Berbagai macam aplikasi diintegrasikan untuk dijalankan pada berbagai komputer. Data yang dihasilkan dari suatu komputer terminal dapat digunakan bersama-sama dengan komputer terminal lainnya.

Kegunaan komputer dapat diperluas dengan sistem komputer dari satu komputer lainnya dapat melakukan pertukaran data, berkomunikasi, penyebaran informasi dan lain-lain.

### 2.2.7 Topologi Jaringan

Menurut (Utomo dwi, 2011), topoologi menggambarkan metode yang digunakan untuk melakukan pengabelan secara fisik dari suatu jaringan. Topologi jaringan adalah susunan atau pemetaan interkoneksi atntara node, dari suatu jaringan, baik secara fisik (rela) dan logis (virtual).

Topologi Jaringan, menurut Sofana (2008) topologi adalah salah satu aturan bagaimana menghubungkan komputer (*node*) satu sama lain secara fisik dan pola hubungan antara komponen-komponen yang berkomunikasi melalui media/peralatanjaringan, seperti: *server, workstation, hub/switch*, dan pemasangan kabel (media transmisi data). Topologi fisik berkaitan dengan bentuk jaringan, seperti bagaimana memilih perangkat dan melakukan instalasi parangkat jaringan. Sedangkan topologi logika berkaitan dengan bagaimana data mengalir di salam topologi fisik.

1. Topologi *bus* disebut juga *linear bus* karena dihubungkan hanya melalui satu kabel yang *linear*, kabel yang umum digunakan adalah kabel koaksial. Semua *node* dihubungkan secara seri menggunakan kabel tersebut.
2. Penempatan kabel yang digunakan dalam ring menggunakan desain yang sederhana. Pada *topologi ring*, setiap komputer terhubung ke komputer selanjutnya, dengan komputer terakhir terhubung ke komuter pertama.
3. Dalam Topologi *star*, semua kabel dihubungkan dari komputer-komputer ke lokasi pusat (*central location*), dimana semuanya terhubung ke suatu alat yang dinakan hub.

4. Topologi *mesh* dapat dikenali dengan hubungan *point-to-point* atau *satu-satu* ke setiap komputer. Setiap komputer terhubung ke komputer lain melalui kabel, bisa menggunakan kabel *coaxial*, *twisted pair*, bahkan serat *optic*. Topologi *mesh* masih cocok digunakan pada jaringan yang sangat kritis.

### 2.2.8 Sistem Operasi

Menurut Sadzali (2012) Sistem Operasi adalah sekumpulan rutin perangkat lunak yang berada di antara program aplikasi dan perangkat keras. Sistem operasi memiliki tugas yaitu mengelola seluruh sumber daya sistem komputer dan sebagai penyedia layanan.

System operasi menyediakan *System Call* (berupa fungsi-fungsi atau *API = Application Programming Interface*). *System Call* ini memberikan abstraksi tingkat tinggi mesin untuk pemrograman. *System Call* berfungsi menghindarkan kompleksitas pemrograman dengan member sekumpulan instruksi yang lebih mudah dan nyaman, system operasi juga sebagai basis untuk program lain dimana program aplikasi dijalankan di atas system operasi, program-program itu memanfaatkan sumber daya sistem komputer dengan cara meminta layanan sistem operasi mengendalikan sumber daya untuk aplikasi sehingga penggunaan sumber daya sistem komputer dapat dilakukan secara benar dan efisien.

Sistem operasi yang dikenal antara lain :

1. *Windows* (95, 98, ME, 2000, XP, VISTA, SERVER, Windows 7, Windows 8)
2. *Linux* (Red Hat, Slackware, Ubuntu, Fedora, Mikrotik, Debian, OpenSUSE)
3. *UNIX*
4. *FreeBSD* (*Barkeley Software Distribution*)
5. *SUN* (*Solaris*)
6. *DOS* (MS-DOS)
7. *Machintosh* (MAC OS, MAC OSX)

#### 2.2.9 MikroTik

Menurut (Siddik., Dkk, 2015) Mikrotik *Router OS* adalah sistem operasi dan perangkat lunak yang dapat digunakan untuk menjadikan komputer menjadi *routernetwork* yang handal, mencakup berbagai fitur yang dibuat untuk ip *network* dan jaringan *wireless*, cocok untuk digunakan oleh ISP dan *Providehotspot*. MikroTik *Router OS*, merupakan sistem operasi Linux *base* yang diperuntukkan sebagai sistem *network router*. Administrasinya bisa dilakukan melalui windows application (WinBox).

```

Welcome to MikroTik Router Software installation

Move around menu using 'p' and 'n' or arrow keys, select with 'spacebar'.
Select all with 'a', minimum with 'm'. Press 'i' to install locally or 'q' to
cancel and reboot.

[X] system          [X] ipv6            [X] routerboard
[X] ppp             [X] isdn           [X] routing
[X] dhcp           [X] kvm            [X] security
[X] advanced-tools [X] lcd            [X] synchronous
[X] arlan          [X] mpls           [X] ups
[X] calea          [X] multicast      [X] user-manager
[X] gps            [X] ntp            [X] wireless
[X] hotspot        [X] radiolan       [X] wireless-nv2

system (depends on nothing):
Main package with basic services and drivers

Do you want to keep old configuration? [y/n]:

```

**Gambar 2.3** Mikrotik Operating OS (Sumber: Jurnal SAINTIKOM Vol. 14 No. 1, Januari 2015)

### 2.2.10 *Open System Interconnection (OSI) Layer*

Menurut (Utomo Dwi, 2011), supaya dapat mengirimkan informasi ke komputer lain, dan dapat menerima dan mengerti informasi, harus ada aturan atau standar untuk proses komunikasi tersebut. Standar ini meyakinkan kita bahwa beberapa jenis produk dan perangkat dapat berkomunikasi dengan perangkat lain yang berbeda melewati beberapa jaringa. Pembakuan standard ini disebut “MODEL”.

Referensi mode jaringan yang sering digunakan adalah *Open System Interconention* (OSI), yang diperkenalkan oleh *International Organisation* (ISO). Referensi model jaringan OSI membagi jaringan komputer menjadi 7 lapisan, setiap lapisan hanya mengatur beberapa layanan dan protocol yang dapat bekerja pada tiap lapisan agar mempermudah pembuatan



program untuk jaringan sehingga rapih dan tidak berantakan serta sulit didefinisikan. Berikut ini adalah penjelasan dari setiap OSI layer.

#### 2.2.10.1 Layer 1 – *Physical Layer*

Lapisan paling bawah dalam model OSI adalah *Physical Layer*. Lapisan ini mengatur bagaimana sinkronisasi pengiriman dan penerimaan data, spesifikasi mekanik, elektrik dan *interface* antar terminal.

Lapisan ini juga berfungsi untuk mendefinikasi media transmisi jaringan, arsitek jaringan, topologi jaringan dan pengabelan. Selain itu, level ini juga mendefinisikan bagaimana *Network Interface Card* (NIC) dapat berinteraksi baik dengan media kabel amupun radio.

#### 2.2.10.2 Layer 2 – *Data link Layer*

Di atas *Physical Layer* adalah *Data-Link Layer*, lapisan dimana data dipersiapkan untuk dikirimkan melalui jaringan, pada lapisan ini paket data di kapsulasi dalam sebuah *frame* (bundle dari data biner) sebelum dikirimkan. *Protocol* pada lapisan ini membantu dalam hal pengalamatan (addressing) dan pendeteksiian kesalahan dari data yang dikirimkan.

*Data-link Layer* terdiri dari dua sublayer yaitu: sublayer *logical Link Control* (LLC) dan sublayer *Media Access Control* (MAC). Sublayer LLC adalah antae muka antara *protocol network layer* dengan metode pengaksesan media midalnya Ethernet atau Token Ring. Sublayer MAC menangani koneksi ke media fisik seperti *twised—pair* atau pengkabelan koaksial.

### 2.2.10.3 Layer 3 – *Network Layer*

Berikutnya adalah *Network Layer* yang bertanggung jawab dalam hal *routing* dari paket-paket data yang didasarkan pada *logical address* dari paket-paket data tersebut. *Network layer* memotong-motong data dan menyusunnya kembali jika diperlukan, ia mengirim paket-paket data dari sumber ke tujuan.

### 2.2.10.4 Layer 4 – *Transport Layer*

Di atas *Network Layer* ada *Transport Layer*, lapisan ini menjamin diterimanya paket data yang dikirim. *Transport layer* juga dapat membentuk sebuah sambungan dan mengirim *acknowledgement* ketika paket data diterima.

### 2.2.10.5 Layer 5 – *Session Layer*

Beikutnya adalah *Session layer*, lapisan ini bertugas untuk mengontrol “dialog” selama komunikasi berlangsung, lapisan ini bertanggungjawab dalam hal bagaimana membentuk sambungan, bagaimana menggunakan sambungan tersebut, dan bagaimana memutuskan sambungan yang terbentuk setelah sesi komunikasi selesai. *Session layer* juga menambahkan *control header* pada paket data selama pertukaran data terjadi.

### 2.2.10.6 Layer 6 – *Presentation Layer*

*Presentation layer* adalah lapisan yang berada di bawah *application layer* dan di atas *session layer*. Lapisan ini menambahkan struktur

pada paket data yang akan dikirimkan. Tugas utama lapisan ini adalah untuk meyakinkan bahwa data atau informasi terkirim dengan bahasa atau syntax yang dapat dipahami oleh *host* yang dituju. *Protocol* pada *presentation layer* dapat menerjemahkan data ke dalam bahasan atau syntax yang dapat dimengerti kemudian mengkomresi atau mengenkripsi data sebelum menyampaikan data ke session layer.

#### 2.2.10.7 Layer 7 – *Application Layer*

Lapisan paling tinggi dari model OSI adalah *application layer*, seluruh layer dibawahnya bekerja untuk lapisan ini, tugas dari application layer mengatur komunikasi antar aplikasi.

Pembentukan paket dimulai dari layer teratas model OSI application layer mengirimkan data ke presentation layer, di presentation layer data ditambahkan *header* dan atau *trailer* kemudian dikirim ke lapisan di bawahnya, pada lapisan di bawahnya pun demikian, data ditambahkan *header* data atau tailer kemudian dikirim ke lapisan di bawahnya lagi, terus demikian sampai ke *physical layer*. Di *physical layer* data dikirimkan melalui media transmisi ke *host* tujuan.

Proses pengiriman paket dari layer ke layer ini disebut dengan “peer-layer communication”. Gambar berikut mengilustrasikan *peer-layer communication*.



**Gambar 2.4** Tugas-tugas OSI Layer

**Sumber:** (<http://lorongbiru.blogspot.com/2010/10/jaringan-model-osi-layer-dan-tcpip>)

### 2.2.11 *Transmission Control Protocol/Internet Protocol (TCP/IP)*

TCP/IP adalah sekumpulan protokol yang didesain untuk melakukan fungsi-fungsi komunikasi data pada internet. TCP/IP terdiri atas sekumpulan protokol yang masing-masing bertanggung jawab atas bagian-bagian tertentu dari komunikasi utama dalam internet serta intranet. protokol ini memungkinkan sistem apapun yang terhubung ke dalamnya

bisa berkomunikasi dengan sistem lain tanpa harus mempedulikan bagaimana *remote system* yang lain tersebut bekerja. Protocol ini dikembangkan pada tahun 1969 oleh DARPA (*Defence Advanced Research Project Agency*) mendanai riset dan pembuatan paket *switching* eksperimental yang diberi nama ARPANET (Sopandi, 2008).

TCP/IP menggunakan model *client-server* dalam berkomunikasi dimana komputer *user (client)* meminta kepada komputer lain dan akan disediakan service tersebut oleh komputer *server*. Beberapa layanan dan *utility* dari TCP/IP meliputi sebagai berikut.

#### 2.2.11.1 *Hypertext Transfer protocol (HTTP)*

HTTP adalah protocol yang dipakai untuk mayoritas komunikasi *World Wide Web (WWW)*. Windows menghadirkan Internet Explorer sebagai client HTTP dan *Internet Information Service (IIS)* sebagai *server* HTTP. Pada umumnya port HTTP adalah 80 (Sopandi, 2008)

#### 2.2.11.2 *Hypertext Transfer protocol Secure (HTTPS)*

HTTPS adalah versi aman dari HTTP, protokol komunikasi dari *World Wide Web*. Ditemukan oleh *Netscape Communication Corporation* untuk menyediakan autentikasi dan komunikasi tersandi dan penggunaan dalam komersi elektrik. Selain menggunakan komunikasi plain text, HTTPS menyandikan data sesi menggunakan protokol SSL (*Secure Socket Layer*) atau protokol TLS (*Transport layer Security*). Kedua protokol tersebut memberikan perlindungan yang memadai dari serangan



*wavesdroppers*, dan *man in the middle attacks*. Pada umumnya *port* HTTPS adalah 443 (Sopandi, 2008).

### 2.2.11.3 Secure Socket Layer (SSL)

SSL adalah protokol yang digunakan untuk *browsing website* secara aman. SSL bertindak sebagai *protocol* yang mengamankan komunikasi antara *client* dan *server*. Protokol ini memfasilitasi penggunaan enkripsi untuk data yang rahasia dan membantu integritas informasi yang dipertukarkan antara *website* dan *web browser*. Pada umumnya *port* SSL sama dengan *port* HTTPS yaitu 443 (Sopandi, 2008)

### 2.2.9.4 Domain Name Service (DNS)

DNS merupakan seperangkat protokol dan layanan pada suatu jaringan TCP/IP yang membolehkan para pemakai jaringan untuk mempergunakan nama-nama hierarki yang sudah dikenal ketika meletakkan *host* ketimbang harus mengingat dan memakai alamat IP-nya. DNS sangat banyak dipakai di internet dan pada kebanyakan perusahaan pribadi dewasa ini. Saat memakai *web browser*, aplikasi *telnet*, utility FTP, atau utility TCP/IP miriip lainnya di Internet, maka *user* mungkin sedang memakaai sebuah *server* DNS. Pada umumnya *port* DNS adalah 53 (Sopandi, 2008).

#### 1. Google *public* DNS

Google *public* DNS adalah alamat DNS yang disediakan oleh google secara gratis. Ketika menggunakan google *public* DNS

pada PC, otomatis alamat DNS dari ISP yang dipakai akan digantikan dengan alamat google *public* DNS. Alamat IP address dari google *public* DNS. Alamat IP address dari google *public* DNS adalah 8.8.8.8 dan 8.8.4.

#### 2.2.11.5 *Simple Network Management Protocol (SNMP)*

SNMP memungkinkan seorang user untuk mengelola node jaringan seperti server, workstation, router, bridge, dan switch dari host sentral. SNMP dapat dipakai untuk mengkonfigurasi device yang jauh, memantau untuk kerja jaringan, mendeteksi kesalahan jaringan atau akses yang tidak cocok, dan mengaudit pemakaian jaringan. (sopandi, 2008)

#### 2.2.12 *IP Address*

Menurut Yudhaswana JoeFRE (2013) *IP Address* atau alamat IP adalah pengenal suatu host dalam suatu jaringan. Pada IP address sendiri terdapat kelas-kelas. Pembagian alamat IP didasarkan pada dua hal, yaitu *network ID* dan *host ID*. *Network ID* digunakan untuk menunjukkan jaringan – jaringan tempat komputer itu berada. Sedangkan *host ID* digunakan sebagai pengenal komputer yang bersifat unik dalam satu jaringan.

Berikut disajikan kelas-kelas alamat IP:

1. Kelas A : Panjang *network* ID adalah 8 bit, panjang host ID adalah 24 bit, digunakan untuk jaringan yang sangat besar. Jumlah host yang dapat ditampung adalah sekitar 16 juta host.
2. Kelas B : Panjang *network* ID adalah 16 bit, panjang host ID adalah 16 bit, diimplementasikan untuk jaringan yang relative besar. Jumlah host yang mampu ditampung adalah 65.532 host.
3. Kelas C : Panjang *network* ID adalah 24 bit, panjang host ID adalah 8 bit. Diimplementasikan untuk jaringan yang relatif besar. Jumlah host yang mampu ditampung adalah 254 host.
4. Kelas D : digunakan untuk keperluan *multicasting*.
5. Kelas E : Alamat IP E tidak digunakan untuk keperluan umum

Ada beberapa aturan yang ditetapkan untuk memberi alamat IP pada suatu host:

1. *Network* ID tidak boleh sama dengan 127 karena nilai ini digunakan untuk *loopback*.
2. *Network* ID dan Host ID tidak boleh sama dengan 0 karena nilai 0 diartikan sebagai alamat jaringan. Pengertian alamat jaringan itu sendiri adalah alamat yang digunakan untuk menunjuk ke suatu host.
3. Host ID harus unik di satu jaringan yang sama

Dalam penelitian yang ditulis oleh (Dwi Utomo, 2011), dituliskan bahwa *IP Address* (alamat IP) adalah identitas khusus yang digunakan untuk

memberikan tanda atau alamat pada sebuah paket data atau pada suatu sistem komputer konsep dasar pengalamatan (*IP address*) di internet adalah awalan (*prefix*) pada *IP Address* dapat digunakan sebagai dasar pengambilan keputusan dalam pemilihan rute paket data ke alamat tujuan. Sebelum memasuki aspek-aspek lebih jauh tentang *IP Address*, penting untuk mengerti lebih dahulu beberapa hal fundamental dari *IP address* itu sendiri.

#### A. Format Alamat IPv4

Untuk IPv4 menggunakan bilangan 32 Bit yang dipisahkan oleh tanda pemisah berupa tanda titik pada setiap bitnya. Tiap 8 bit ini disebut sebagai octet. Pengalamatan IP berupa nomor 32 bit tersebut terdiri dari *network ID* dan *host ID*. *Network ID (NetID)* menunjukkan nomor jaringan sedangkan *host ID* mengidentifikasi host dalam satu jaringan.

Desimal	167	205	206	100
Biner	10100111	11001101	11001110	01100100

#### B. Kelas Alamat IP

Jika dilihat dari bentuknya, *IP Address* terdiri dari 4 buah bilangan oktat (8 bit). Nilai terbesar dari bilangan biner 8 bit yaitu  $255 = (2^7 + 2^6 + 2^5 + 2^4 + 2^3 + 2^2 + 2^1 + 2^0)$ . Jumlah keseluruhan *IP Address* adalah sebanyak  $255 \times 255 \times 255 \times 255$ .

*IP Address* sebanyak inilah yang harus dibagikan ke seluruh pengguna jaringan internet ke seluruh dunia. Untuk mempermudah

membagikannya, *IP Address* dikelompokkan dalam kelas-kelas, hal ini dilakukan untuk memudahkan pendistribusian pendaftaran *IP Address*.

Kelas-kelas *IP address* yang umum digunakan terdiri dari:

1. Kelas A: 1-126, dengan batasan 1.0.0.0 – 126.255.255.254

Dengan *Broadcast Address* 126.255.255.255

*Subnet Mask*-nya : 255.0.0.0

2. Kelas B : 128-191, dengan batasan 128.0.0.0-191.255.255.254

Dengan *broadcast address* 192.255.255.255

*Subnet mask*-nya: 255.255.0.0

3. Kelas C: 192-223, dengan batasan 192.0.0.0-223.255.255.254

Dengan *broadcast address* 223.255.255.0

*Subnet Mask*-nya : 255.255.255.0

Dalam pemberian *IP address* juga harus berdasarkan aturan-aturan dasar, seperti *network ID* dan *host ID* tidak boleh bernilai 0 (nol) atau 255 (semua bit di set1). *Network ID* juga tidak boleh bernilai 127, karena *default* sudah digunakan sebagai IP lokal (*loppback IP address*)



### 2.2.13 Internet

Dalam penelitian yang dilakukan oleh (Rofiq Muhamad, 2013) dituliskan bahwa *Internet* merupakan singkatan dari *Interconnect Networking*, yang berarti suatu jaringan komputer yang terhubung dengan luas. Internet berasal dari sebuah jaringan komputer yang terhubung dengan luas. Internet berasal dari sebuah jaringan komputer yang dibuat pada tahun 1970-an yang harus terus berkembang sampai sekarang menjadi jaringan dunia yang sangat luas. Jaringan tersebut diberi nama arpanet, yaitu jaringan yang dibentuk oleh Departemen Pertahanan Amerika Serikat. Kemudian, jaringan komputer tersebut diperbaharui dan dikembangkan sampai sekarang dan menjadi tulang punggung global untuk sumber daya informasi yang disebut internet .

Jenis Koneksi Internet yaitu sebagai berikut:

1. *User* pribadi: mempergunakan koneksi dial-up modem (menggunakan line telepon).
2. *User* Institusi/corporate : Koneksi dial-up Analog/Digital (ISDN), koneksi leased-line (permanen), dan koneksi VSAT (*Verry Small Arpperture Terminal*).

### 2.2.14 Quality of Service (QoS)

Dalam penelitian yang dilakukan oleh (Mandarani & Zaini, 2015) yang berjudul Pengembangan Sistem Monitoring pada *Building Automation*

*System (BAS) Berbasis Web* di Fakultas Teknik Universitas Andalas, menyebutkan bahwa *QoS (Quality of Service)* adalah kemampuan suatu jaringan untuk menyediakan layanan yang baik dengan menyediakan *bandwidth*, mengatasi *jitter* dan *delay*. Performansi merupakan kumpulan dari beberapa parameter besaran teknis, salah satu parameter tersebut adalah *delay (latency)*, yaitu waktu yang dibutuhkan data untuk menempuh jarak dari asal ke tujuan. Delay dapat dipengaruhi oleh jarak, media fisik, kngesti atau juga waktu proses yang lama.

Sedangkan menurut (silitoga & Sri Morina, 2014) *QoS (Quality of Service)* merupakan teknologi yang diterapkan dalam jaringan komputer untuk memberikan layanan yang optimal dan adil bagi para pengguna jaringan komputer. QoS memungkinkan administrator jaringan untuk dapat menangani berbagai efek akibat terjadinya kemacetan (*congestion*) pada lalu lintas aliran paket di dalam jaringan.

Kinerja jaringan komputer dapat bervariasi akibat beberapa masalah, seperti halnya masalah *bandwidth*, *latency* dan *jitter*, yang dapat membuat efek yang cukup besar bagi banyak aplikasi dalam satu jaringan.

Baik atau buruknya kinerja sebuah jaringan data diukur dengan menggunakan *QoS (Quality of Service.)* *QoS (Quality of Service)* merupakan teknologi yang diterapkan dalam jaringan komputer untuk memberikan layanan yang optimal dan adil bagi para pengguna jaringan komputer. QoS memungkinkan administrator jaringan untuk dapat menangani berbagai efek

akibat terjadinya kemacetan (*congestion*) pada lalu lintas aliran paket di dalam jaringan

### 2.2.15 Perangkat Jaringan

Dalam penelitian yang ditulis oleh (Dwi Utomo, 2011) Perangkat keras (*hardware*) adalah semua bagian fisik komputer, dan dibedakan dengan data yang berada di dalamnya atau yang beroperasi di dalamnya, dan dibedakan dengan perangkat lunak (*software*) yang menyediakan instruksi untuk perangkat keras dalam menyelesaikan tugasnya.

Secara umum, perangkat keras yang dibutuhkan untuk membangun sebuah jaringan komputer yaitu : Komputer, *Network Internet Card* (NIC), *Switch*, dan segala sesuatu yang berhubungan dengan koneksi jaringan seperti : *Bridges*, *Router* dan lainnya yang dibutuhkan untuk proses transformasi data di dalam jaringan.

#### 2.2.15.1 *Personal Computer (PC)*

Personal Computer atau PC merupakan perangkat utama dalam suatu jaringan komputer. PC ini lah yang akan bekerja mengirim dan mengakses data dalam jaringan. Kemampuan suatu PC sangat menentukan sekali unjuk kerja dalam jaringan. Semakin tinggi kemampuan suatu PC, maka akses yang akan dilakukan pun akan semakin cepat.

#### 2.2.15.2 *Network Internet Card (NIC)*

Kartu jaringan merupakan perangkat yang menyediakan media untuk menghubungkan antarkomputer. Kebanyakan kartu jaringan adalah kartu internal, yaitu kartu jaringan yang dipasang pada slot ekspansi di dalam komputer. Kartu jaringan umumnya telah menyediakan port koneksi untuk kabel koaksial ataupun kabel *Twisted pair*.

#### 2.2.15.3 *Switch*

*Switch* adalah sebuah alat jaringan yang melakukan *bridging* transparan (penghubung segmentasi banyak jaringan dengan *forwarding* berdasarkan *MAC address*).

*Switch* jaringan dapat digunakan sebagai penghubung komputer atau *router* pada satu area yang terbatas, *switch* juga bekerja pada lapisan *data link*, cara kerja *switch* hampir sama seperti *bridge*, tetapi *switch* sejumlah *port* sehingga sering dinamakan *multiport bridge*.

#### 2.2.15.4 *Router*

*Router* adalah sebuah alat jaringan komputer yang mengirimkan paket data melalui sebuah jaringan atau internet menuju tujuannya, melalui sebuah proses yang dikenal sebagai *routing*. Proses

*routing* terjadi pada lapisan 3 (*Network Layer* seperti pada *Internet Protokol*) dari tujuh lapisan OSI.

*Router* berfungsi sebagai penghubung antar dua atau lebih jaringan untuk meneruskan data dari satu jaringan ke jaringan lainnya. *Router* berbeda dengan *switch*. *Switch* merupakan penghubung beberapa alat untuk membentuk suatu *Local Area network* (LAN).

*Router* sangat banyak digunakan dalam jaringan berbasis *teknologi protocol TCP/IP*, dan *router* jenis itu disebut juga dengan *IP router*. *internet* merupakan contoh utama dari sebuah jaringan yang memiliki banyak *router IP*. *Router* dapat digunakan untuk menghubungkan banyak jaringan kecil ke sebuah jaringan yang lebih besar, yang disebut dengan *internetwork*, atau untuk membagi sebuah jaringan besar ke dalam beberapa *subnetwork* untuk meningkatkan kinerja dan juga mempermudah manajemennya.

*Router* juga kadang digunakan untuk mengkoneksikan dua buah jaringan yang menggunakan media yang berbeda, seperti halnya *router wireless* yang pada umumnya selain ia dapat menghubungkan komputer dengan menggunakan radio, ia juga mendukung penghubungan komputer dengan kabe UTP.

#### 2.2.15.5 *Modem Router*



*Modem Router* dalam piranti *router* yang sudah *include* dengan modem HSDA.. teknologi ini mungkin saja sudah lama dikembangkan, namun di Indonesia sendiri baru diperkenalkan awal tahun 2011.

#### 2.2.15.6 3G Wireless Router

Perangkat keras ini sebenarnya sama dengan modem *router*, tetapi perbedaan yang mendasar adalah bahwa perangkat ini menggunakan modem usb tambahan sebagai koneksi internet.

Sebagai contohnya adalah TP-LINK MR3420, yang mempunyai slot usb dan *port* ADSL. *Port* USB ini dapat digunakan untuk modem usb GSM maupun CDMA.

Kendala dari perangkat keras ini adalah bahwa tidak semua mode usb akan terdeteksi, jadi *firmware* dari perangkat ini harus di *update* terlebih dahulu.

#### 2.2.16 Subnetting

Menurut (Dwi Utomo, 2011) Sebuah jaringan dapat dipecahkan menjadi beberapa jaringan baru, proses ini disebut *subnetting*. Tujuan dari *subnetting* yaitu untuk untuk mereduksi trafik jaringan, mengoptimalkan *performace* jaringan yang dibatasi area geografis luas. Melalui *subnetting* sebuah alamat jaringan (*network address*) tunggal dipecah menjadi *subnetwork* atau disingkat *subnet*. Sebagai contoh *network* 192.168.10.0,

168.20.0.0 dan 192.168.30.0 merupakan *subnet network* tunggal 192.168.8.0.

*Subnet address* dibuat dengan meminjamkan *bit* porsi *host* dan menjadikannya sebagai *subnet*. Jumlah *bit* yang dipinjam bervariasi tergantung pada nilai *subnet mask*.

### 2.2.17 *Network Address Translation (NAT)*

*Network Address translation* atau yang lebih biasa disebut dengan NAT adalah suatu metode untuk menghubungkan lebih dari satu komputer ke jaringan internet dengan menggunakan satu alamat IP *Public*. Banyaknya penggunaan metode ini disebabkan karena ketersediaan alamat IP yang terbatas, kebutuhan akan keamanan (*security*), dan kemudahan serta fleksibilitas dalam administrasi jaringan.

Saat ini, Protokol IP yang banyak digunakan adalah IP versi 4 (IPv4). Dengan panjang alamat 4 *byte* berarti terdapat  $2^{32} = 4.294.967.296$  alamat IP yang tersedia. Jumlah ini secara teoritis adalah jumlah komputer yang dapat langsung koneksi ke internet. Karena keterbatasan inilah sebagian besar ISP (*Internet Service Provider*) hanya akan mengalokasikan satu alamat untuk satu pengguna dalam alamat untuk satu pengguna dan alamat ini bersifat dinamik, dan arti alamat IP yang diberikan akan berbeda setiap kali *user* melakukan koneksi ke internet. (Nugroho, 2005 – Dwi Utomo, 2011). Hal ini akan menyulitkan untuk bisnis golongan menengah ke bawah di satu sisi mereka membutuhkan banyak komputer yang terkoneksi ke internet, akan tetapi di sisi lain hanya tersedia satu alamat IP yang berarti hanya ada satu komputer yang bisa terkoneksi ke internet hal ini bisa

diatasi dengan metode NAT. dengan NAT *gateway* yang dijalankan di salah satu komputer, satu alamat IP tersebut dapat dibagi ke beberapa komputer yang lain dan mereka bisa melakukan koneksi ke internet secara bersamaan.

Dengan NAT, suatu jaringan yang besar data dipecah-pecah menjadi jaringan yang lebih kecil. Bagian-bagian kecil tersebut masing-masing memiliki satu alamat IP, sehingga dapat menambahkan atau mengurangi jumlah komputer tanpa mempengaruhi jaringan secara keseluruhan.

Dalam mikrotik, NAT dikenal juga sebagai *masquerade* yaitu fasilitas *router* untuk meneruskan paket dari IP asal dan atau ke IP tujuan.

Terdapat dua jenis NAT, yaitu *source NAT* (*srcnat*) dan *destination NAT* (*dstnat*).

### 2.2.18 Winbox

Winbox adalah sebuah utility yang digunakan untuk melakukan remote ke server mikrotik dalam mode GUI. Jika untuk mengkonfigurasi mikrotik dalam text mode melalui PC itu sendiri, maka GUI yang menggunakan winbox ini kita konfigurasi mikrotik melalui komputer client.

Mengkonfigurasi mikrotik melalui winbox ini lebih banyak digunakan karena selain penggunaannya yang mudah, kita juga tidak harus menghafal perintah-perintah *console*.

Untuk mendapatkan winbox, kita dapat mengunduhnya atau bisa juga didaatkan di mikrotik. Dengan cara membuka browser komputer client yang telah tersambung dengan mikrotik, lalu tuliskan di address bar <http://ipaddressrouter/winbox/winbox.exe>. Atau bisa mengunduhnya di website <http://www.mikrotik.co.id/download.php>