

BAB 2

DASAR TEORI

2.1 STUDI PUSTAKA

Tabel 2.1 Studi Pustaka sebagai acuan teori-teori yang digunakan pada penelitian tugas akhir ini. Studi pustaka berupa jurnal atau tugas akhir yang berkaitan dengan *Queue Tree* dan QoS.

Tabel 2.1 Studi Pustaka

No	Jurnal/Tugas Akhir	Keterangan
1	J. Penelitian, I. Komputer, A. Budiman, "MANAJEMEN BANDWIDTH SIMPLE QUEUE DAN QUEUE TREE PADA PT . ENDORSINDO MAKMUR SELARAS," vol. 3, no. 1, pp. 11–27, 2015.[1]	Melakukan pengelolaan <i>bandwidth</i> menggunakan metode <i>Queue Tree</i> maka pembagian <i>bandwidth</i> pada setiap pengguna dapat sesuai dengan kebutuhannya, tanpa adanya pemakaian <i>bandwidth</i> yang terlalu besar oleh salah satu pengguna saja.
2	Mirsantoso, T. Kalsum Umi, Supardi, Reno, "Implementasi Dan Analisa Per Connection Queue (Pcq) Sebagai Kontrol Penggunaan Internet Pada Laboratorium Komputer," <i>Media Infotama</i> , vol. 11, no. 2, pp. 139–148, 2015.[2]	Dengan menggunakan manajemen bandwith dengan tipe per connection queue ini, dapat membatasi penyedotan bandwith oleh download manajer seperti internet download manajer (IDM) dan sejenisnya.
3	Pamungkas, C. A. (2016). <i>Manajemen Bandwith Menggunakan Mikrotik Routerboard di Politeknik Indonusa Surakarta</i> . Jurnal INFORMA Politeknik	Semua device yang terhubung dengan jaringan dapat menggunakan internet dengan lancar dan stabil walaupun semua unit menggunakan

	Indonusa Surakarta ISSN : 2442-7942 Tahun 2016, Vol. 1 Nomor 3.[9]	internet dalam waktu yang bersamaan.
--	---	--------------------------------------

2.2 MIKROTIK

2.2.1 Pengertian Mikrotik

Mikrotik adalah sebuah perusahaan yang bergerak di bidang produksi perangkat keras (*hardware*) dan perangkat lunak (*software*). Mikrotik dibuat oleh MikroTikls, sebuah perusahaan yang berhubungan dengan sistem jaringan komputer yang berkantor pusat di Kota Riga, Latvia, bersebelahan dengan Rusia. MikroTikls (dengan *trade name* Mikrotik®) didirikan pada tahun 1995 bertujuan mengembangkan *router* dan sistem ISP (*Internet Service Provider*) *nirkabel* untuk jalur *data internet* di banyak negara, antara lain Iraq, Sri Lanka, Ghana dan banyak negara lainnya. Mikrotik saat ini menyediakan *hardware* dan *software* untuk konektivitas *internet* di sebagian besar negara di seluruh dunia termasuk Indonesia.



Gambar 2.1 Logo Mikrotik[4]

Mikrotik Indonesia, di operasikan oleh Citraweb Nusa Infomedia sejak tahun 2001, dan telah menjadi *reseller* resmi Mikrotik sejak tahun 2002. Selain mengelola produk Mikrotik, Citraweb juga mengoperasikan sebuah *wireless* ISP dengan nama Citra-Net. Produk *hardware* unggulan Mikrotik berupa RouterBoard, sedangkan produk *software* unggulan Mikrotik adalah Mikrotik RouterOS (*Operating System*).[4]

2.2.2 Mikrotik RouterOS

MikroTik RouterOS adalah sistem operasi dan perangkat lunak yang dapat digunakan untuk menjadikan sebuah komputer menjadi suatu *router network* yang handal, mencakup berbagai fitur yang dibuat untuk *Internet Protocol (IP) network* dan jaringan *wireless*, sehingga cocok digunakan oleh ISP dan *provider hotspot*. Untuk

instalasi Mikrotik tidak dibutuhkan piranti lunak tambahan atau komponen tambahan lain.

```

MMM      MMM      KKK      TTTTTTTTTT      KKK
MMMM     MMMM     KKK      TTTTTTTTTT      KKK
MMM MMMM  MMM  III  KKK  KKK  RRRRRR  000000  TTT  III  KKK  KKK
MMM  m  MMM  III  KKKK  RRR  RRR  000  000  TTT  III  KKKK
MMM      MMM  III  KKK  KKK  RRRRRR  000  000  TTT  III  KKK  KKK
MMM      MMM  III  KKK  KKK  RRR  RRR  000000  TTT  III  KKK  KKK

MikroTik RouterOS 4.16 (c) 1999-2010      http://www.mikrotik.com/

=====
ROUTER HAS NO SOFTWARE KEY
-----
You have 23h19m to configure the router to be remotely accessible,
and to enter the key by pasting it in a Telnet window or in Winbox.
See www.mikrotik.com/key for more details.

Current installation "software ID": IMH7-006K
Please press "Enter" to continue!

[admin@MikroTik] > interface print
Flags: D - dynamic, X - disabled, R - running, S - slave
#      NAME      TYPE      MTU  L2MTU
0  R  ether1      ether    1500
1  R  ether2      ether    1500
[admin@MikroTik] > =

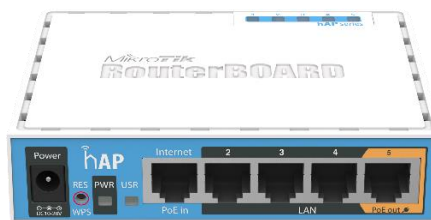
```

Gambar 2.2 Mikrotik OS

Mikrotik didesain untuk mudah digunakan dan sangat baik digunakan untuk keperluan administrasi jaringan komputer seperti merancang dan membangun sebuah sistem jaringan komputer skala kecil hingga yang kompleks sekalipun.

2.2.3 Mikrotik RouterBoard

Mikrotik RouterBoard adalah suatu *hardware* yang digunakan untuk menghubungkan beberapa jaringan (*network*) tanpa perlu di *install* ke sebuah Personal Computer (PC). RouterBoard seperti sebuah PC *mini* yang terintegrasi karena dalam satu *board* tertanam *processor*, ram, rom, dan *memory flash*. RouterBoard menggunakan sistem operasi RouterOS yang berfungsi sebagai *router* jaringan, *Bandwidth management*, *Proxy server*, *Dynamic Host Configuration Protocol* (DHCP), *Domain Name System* (DNS) *server* dan bisa juga berfungsi sebagai *Hotspot server*. Ada beberapa seri RouterBoard yang juga bisa berfungsi sebagai *Wifi Access Point* (AP), *Bridge*, *Wireless Distribution System* (WDS) ataupun sebagai *Wifi client*. Sebagian besar ISP *wireless* menggunakan RouterBoard untuk menjalankan fungsi *wireless*nya baik sebagai AP ataupun *Client*. Dengan RouterBoard bisa dijalankan fungsi sebuah *router* tanpa tergantung pada PC lagi, karena semua fungsi pada *router* sudah ada dalam RouterBoard.



Gambar 2.3 MikroTik-RouterBoard-RB951.[4]

Jika dibandingkan dengan PC yang diinstal RouterOS, RouterBoard ukurannya lebih kecil, lebih kompak dan hemat listrik karena hanya menggunakan adaptor sebagai sumber tenaga inputan. Contoh aplikasi yang dapat diterapkan dengan adanya Mikrotik RouterBoard selain *routing* adalah aplikasi manajemen *bandwidth*, *firewall*, *wireless access point*, *backhaul link*, *system hotspot*, *Virtual Private Network (VPN)* *server* dan masih banyak lagi.[4]

2.3 QUALITY OF SERVICE (QOS)

Quality Of Service adalah kemampuan untuk menyediakan jaminan dan performa layanan pada suatu jaringan. *QoS* digunakan untuk mengukur tingkat kualitas koneksi jaringan *Transmission Control Protocol/Internet Protocol (TCP/IP)* *internet* atau intranet. Terdapat banyak hal yang bisa terjadi pada paket ketika mereka melakukan perjalanan dari asal ke tujuan, masalah-masalah tersebut sering disebut sebagai parameter-parameter *QoS*, yang meliputi :

2.3.1 Throughput

Throughput adalah jumlah total kedatangan paket yang sukses yang diamati selama *interval* waktu tertentu. Hal ini menentukan besarnya trafik yang dapat diperoleh aplikasi saat melewati suatu jaringan.[5]

Tabel 2.2 *Standard Baku Throughput*[6]

Kategori	Throughput (Mbps)	Index
Terbaik	> 2,1	4
Lebih Baik	1,2 – 2,1	3
Baik	0,7 – 1,2	2
Cukup Baik	0,338 – 0,7	1
Buruk	0 – 0,338	0

2.3.2 Delay

Delay merupakan waktu yang dibutuhkan oleh *data* untuk menempuh jarak dari asal ke tujuan. *Delay* dari pengirim ke penerima pada dasarnya tersusun atas *hardware latency*, *delay* akses, dan *delay* transmisi. *Delay* yang paling sering dialami oleh trafik yang lewat adalah *delay* transmisi.

Tabel 2.3 *Standard Baku Delay*[6]

Kategori	Besar Delay	Indeks
Sangat Bagus	<150 ms	4
Bagus	150 – 300 ms	3
Sedang	300 – 450 ms	2
Jelek	>450 ms	1

2.3.3 Packet Loss

Packet loss merupakan suatu parameter yang menggambarkan suatu kondisi yang menunjukkan jumlah total paket yang hilang diakibatkan *collision* dan *congestion*. Hilangnya paket tersebut dapat disebabkan oleh beberapa kemungkinan, diantaranya yaitu :

1. Terjadinya *overload* trafik didalam 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*.

Tabel 2.4 *Standard Baku Packet loss* [6]

Kategori	Packet loss (%)	Index
Sangat Bagus	0-2	4
Bagus	3-14	3
Sedang	15-24	2
Jelek	25	1

2.3.4 Jitter

Jitter merupakan variasi – variasi dalam panjang antrian, dalam waktu pengolahan *data* dan juga dalam waktu penghimpunan ulang paket – paket perjalanan akhir. Besarnya nilai *jitter* akan sangat dipengaruhi oleh variasi beban trafik dan besarnya tumbukan antar paket (*congestion*) yang ada dalam jaringan IP. Semakin besar beban trafik di dalam jaringan akan menyebabkan semakin besar pula peluang terjadinya *congestion* dengan demikian nilai *jitter*-nya akan semakin besar. Semakin besar nilai *jitter* akan mengakibatkan nilai *QoS* akan semakin turun. Untuk

mendapatkan nilai *QoS* jaringan yang baik, nilai *jitter* harus dijaga seminimum mungkin. Terdapat empat kategori penurunan performansi jaringan berdasarkan nilai *peak jitter* sesuai dengan versi TIPHON (Joesman, 2008 dalam Fatoni, 2011), yaitu :

Tabel 2.5 *Standard Baku Jitter*[6]

Kategori Degradasi	Peak Jitter	Indeks
Sangat Bagus	0 ms	4
Bagus	0 - 75 ms	3
Sedang	75 - 125 ms	2
Jelek	125 - 225 ms	1

2.4 Manajemen *Bandwidth*

Bandwidth merupakan lebar pita frekuensi yang digunakan untuk transmisi data dalam medium transmisi elektronik pada suatu komunikasi data. *Bandwidth* juga bisa berarti jumlah konsumsi paket *data* per satuan waktu dinyatakan dengan satuan *bit per second* (bps). *Bandwidth* merupakan besaran yang paling banyak digunakan sebagai ukuran kecepatan aliran *data*. *Bandwidth internet* disediakan oleh *provider internet* dengan jumlah tertentu tergantung sewa pelanggan. *Bit* atau *binary digit* merupakan basis *data* yang terdiri dari angka 0 dan 1. Satuan ini menggambarkan seberapa banyak *bit* (0 dan 1) yang dapat mengalir dari satu tempat ke tempat yang lain dalam satu detiknya melalui perantara suatu media. Di dunia komunikasi *data* terutama dalam jaringan TCP/IP, *bandwidth* adalah sumber daya utama yang harus digunakan secara efisien. Dalam rangka pemanfaatan *bandwidth* secara optimal dan efisien maka diperlukan teknik khusus untuk manajemen sumber daya tersebut. Sehingga dengan manajemen *bandwidth*, sumber daya jaringan bisa dikendalikan dan dapat memenuhi kriteria *Quality of Service* (QoS).

Manajemen *bandwidth* adalah pengaturan dalam pembagian alokasi *bandwidth* secara efektif sesuai dengan kebutuhan yang diharapkan. Penggunaan *internet* secara bersamaan dapat mempengaruhi *bandwidth* dan *transfer data* antar komputer, oleh karena itu dibutuhkan manajemen *bandwidth*. Setiap *user* mempunyai kebutuhan masing-masing dalam mengakses *internet*, ada yang menghabiskan *bandwidth* dengan men-download MP3, *video* atau *streaming*, ada yang *chatting* atau yang sekedar

mengecek *email*. Hal inilah yang menyebabkan pentingnya pengalokasian *bandwidth* agar penggunaannya bisa terasa adil, atau bisa disebut juga *fair usage*.

Manajemen *bandwidth* bisa dianalogikan dengan penggunaan pipa yang mengalirkan air ke seluruh penggunanya. Setiap pengguna tentu tidak dapat sepenuhnya mendapatkan air yang dialirkan dari sumbernya (*internet*), hal ini tergantung dengan jumlah *bandwidth* yang tersedia, pembatasan tersebut adalah nilai maksimal yang didapatkan oleh setiap pengguna, dan sewaktu-waktu dapat *menurun* sesuai dengan keadaan serta kondisi kesibukan pertukaran *data*. Ketika banyak orang yang mengirim dan menerima *email*, berkomunikasi dengan *video streaming*, *men-download data* dengan ukuran yang besar sehingga dapat mempengaruhi *throughput*, yaitu jumlah paket yang terkirim dalam satuan waktu.[4]

Istilah *bandwidth management* sering dipertukarkan dengan istilah *traffic control*, yang dapat didefinisikan sebagai pengalokasian yang tepat dari suatu *bandwidth* untuk mendukung kebutuhan atau keperluan aplikasi atau suatu layanan jaringan. Dalam mengendalikan trafik, administrator jaringan bisa memilih beberapa metode tergantung dari situasi pada jaringan LAN atau *backbone*. Tiap trafik akan dikendalikan dengan metode tertentu yang akan berdampak pada kecepatan akses. Berikut yang termasuk dalam teknik pengendalian trafik disiplin antrian adalah :

1. *Scheduler* : Menggunakan algoritma *reschedule* dan *men-drop* paket yang tidak muat dalam *interface*. Contohnya adalah *Packets First-In First-Out (PFIFO)*, *Bytes First-In First-Out (BFIFO)*, *Stochastic Fairness Queuing (SFQ)*, *Random Early Detect (RED)*.
2. *Shaper* : Berguna untuk membatasi kecepatan akses paket. Contohnya adalah metode *Queue Tree*.

2.4.1 Simple Queue

Simple queue adalah cara termudah untuk melakukan *limit bandwidth* yang dapat digunakan untuk membatasi *bandwidth* berdasarkan alamat IP tertentu. *Queue simple* dapat digunakan untuk membangun aplikasi QoS yang lebih rumit. Fitur yang dimiliki *simple queue* antara lain:

1. Mengizinkan pembuatan aturan *queue* dengan pemilihan interval waktu.
2. Penggunaan prioritas.
3. Menggunakan *multiple packet*.

4. *Limit* trafik dari dua arah (satu *limit* untuk total *upload* + *download*).[7]

2.4.2 *Queue Tree*

Metode *queue tree* merupakan aturan yang diterapkan dalam pembagian *bandwidth* pada mikrotik. *Mangle* merupakan konfigurasi yang berguna untuk menandai aliran paket kemudian menggunakan tanda ini sebagai pengidentifikasian arus pada paket *queue tree*. *Queue tree* merupakan *limit bandwidth* yang cukup kompleks karena pembatasan dapat dikelompokkan berdasarkan protokol, *port* atau kelompok IP *Address*. Sebelum melakukan pembatas harus ditandai aliran paket menggunakan suatu *mangle* (istilah pada mikrotik) agar paket tersebut dapat dikenal oleh *queue tree* hal ini bertujuan untuk membedakan paket yang *downlink only* atau *uplink only* sehingga *limit* pada *bandwidth* dapat bekerja optimal

Queue Tree hanya menciptakan satu arah antrian pada salah satu HTB. Hal ini juga satu-satunya cara bagaimana cara menambahkan antrian pada antarmuka yang terpisah. Dengan cara ini sangat mungkin untuk memudahkan konfigurasi *mangle*, tidak perlu menandai yang terpisah untuk *download* dan *upload*, *upload* hanya akan mendapatkan *interface public* dan *download* hanya akan mendapatkan *interface private*. Itu juga dimungkinkan untuk memiliki antrian ganda (contoh: prioritas lalu lintas pada *global-in* atau *global-out*, pembatasan per klien pada *interface* yang keluar).[7]

Queue Tree berfungsi untuk memlimit *bandwidth* pada mikrotik yang mempunyai 2 koneksi internet karna packet marknya lebih berfungsi dari pada di Penjelasan beberapa argumen di *Queue Tree* :

1. *Parent* : berguna untuk menentukan apakah *queue* yang dipilih bertugas sebagai *child queue* ada beberapa pilihan default di parent *queue tree* yang biasanya digunakan untuk induk *queue*.
2. *Global-in* : Mewakili semua *input interface* pada umumnya. Maksudnya disini *interface* yang menerima input data/trafik sebelum difilter seperti trafik *upload*.
3. *Global-out* : Mewakili semua *output interface* pada umumnya. Maksudnya disini *interface* yang mengeluarkan output data/trafik yang sudah difilter seperti trafik *download*.

4. *Global-total* : Mewakili semua *input* dan *output interface* secara bersama, dengan kata lain merupakan penyatuan dari *global-in* dan *global-out*.
5. *<interface name>*: ex: LAN atau WAN : Mewakili salah satu *interface* keluar. Maksudnya disini hanya trafik yang keluar dari *interface* ini yang akan di *queue*.
6. *Packet Mark* : Digunakan untuk menandai paket yang sudah ditandai di */ip firewall mangle*. *Priority (1 s/d 8)* : Digunakan untuk memprioritaskan *child queue* dari *child queue* lainnya. *Priority* tidak bekerja pada induk *queue*. *Child Queue* yang mempunyai *priority* satu akan mencapai *limit-at* lebih dulu dari pada *child queue* yang ber*priority*.
7. *Queue Type* : Digunakan untuk memilih tipe *queue* yang bisa dibuat secara khusus dibagian *queue types*.
8. *Limit At* : *Bandwidth* minimal yang diperoleh oleh target/ip yang di *queue*.
9. *Max Limit* : *Bandwidth* maksimal yang bisa dicapai oleh target/ip yang di *queue*.
10. *Burst limit* : *Bandwidth* maksimal yang bisa dicapai oleh target/ip yang di *queue* ketika *burst* sedang aktif
11. *Burst time* : Periode waktu dalam detik, dimana data *Rate* rata2 di kalkulasikan.
12. *Burst Threshold* : Digunakan ketika data *Rate* dibawah nilai *burst threshold* maka *burst* diperbolehkan. Ketika data *Rate* sama dengan nilai *burst threshold* *burst* dilarang. Untuk mengoptimalkan *burst* nilai *burst threshold* harus diatas nilai *Limit At* dan dibawah nilai *Max Limit*

2.4.3 Mangle

Mangle pada mikrotik merupakan suatu cara untuk menandai paket data dan koneksi tertentu yang dapat diterapkan pada fitur mikrotik lainnya, seperti pada *routes*, pemisahan bandwidth pada *queues*, NAT dan filter *rules*. Tanda *mangle* yang ada pada router mikrotik hanya bisa digunakan pada router itu sendiri. Dan yang perlu diingat bahwa proses pembacaan rule *mangle* ini dilakukan dari urutan pertama ke bawah. *Mangle* digunakan untuk menandai paket data tertentu, pekerjaan menandai ini biasa di sebut *marking*. Ada 3 jenis *marking* yang bisa digunakan di Mikrotik yaitu *Connection-Mark*, *Packet-Mark* dan *Route-Mark*. [7]

1. *Connection-Mark*

Connection-Mark ini *marking* yang digunakan untuk menandai 1 koneksi baik *request* maupun *response*. *Mark-Connection* akan menandai *packet* pertama yang melewati *router* kemudian paket dibelakangnya akan menerima tanda yang sama. *mark* ini akan menandai *packet Request dan Response* , maka secara otomatis paket *Response* dari Internet juga akan mendapat *marking* yang sama.

2. *Packet-Mark*

Packet-Mark ini adalah *marking* yang digunakan untuk menandai setiap paket yang melewati *router*, sama seperti *Connection Mark* , *marking* ini juga menandai *traffic Request* maupun *Response*. setiap *packet* akan ditandai satu persatu. Karena harus bertanya dan menandai setiap *packet* yang melewati *Router* maka *mark-packet* ini akan membutuhkan Lebih banyak *Resource*, baik itu *packet request* ataupun *response router* akan tetap memeriksa setiap *packet* tersebut untuk ditandai.

3. *Route-Mark*

Route-Mark ini digunakan untuk pemilihan jalur *routing* , semisal kita menggunakan 2 ISP maka kita bisa menentukan ISP mana yang akan digunakan setiap client menggunakan *marking* ini.

2.5 Model Referensi *Open System Interconnection* (OSI)

Open System Interconnection merupakan sebuah protokol interkoneksi sistem terbuka yang dibuat oleh *International Organization for Standardization* untuk menyediakan model dasar sehingga dapat memodelkan semua protokol. Model OSI telah dipergunakan dalam praktik. Namun, model ini umumnya berfungsi sebagai prototipe teoritis berupa grafik dan blok-blok diagram untuk membuat sistem protokol jaringan yang baik. Protokol-protokol jaringan yang ada saat ini mengambil sebagian atau keseluruhan fungsi dasar pada model OSI.

2.5.1 karakteristik OSI Layer

Model OSI terbagi menjadi tujuh lapisan dan setiap lapisan memiliki fungsi masing-masing. Lapisan-lapisan ini merepresentasikan tipe-tipe fungsi yang seharusnya didukung oleh protokol. Lapisan osi disusun berdasarkan blok fungsi model logik, dari atas ke bawah. Bagian atas lapisan merupakan lapisan yang dekat

dengan pengguna atau aplikasi, sedangkan lapisan bagian bawah memiliki fungsi yang lebih dekat dengan fisik atau antarmuka jaringan.

2.5.2 Lapisan-lapisan Model OSI

Model OSI memiliki tujuh lapisan, diantaranya :

1. Lapisan Fisik (*Physical Layer*)

Lapisan fisik merupakan lapisan pertama, atau lapisan terbawah pada model OSI. Lapisan ini berfungsi untuk mengirim dan menerima *data* dari atau ke media fisik seperti konektor, kabel, perangkat keras dan media radio maupun satelit.

2. Lapisan Hubungan *Data* (*Data Link Layer*)

Lapisan hubungan *data* merupakan lapisan kedua. Awalnya lapisan ini dibuat sebagai lapisan fungsional tunggal (*single function layer*). Namun karena kebutuhan yang semakin banyak, sehingga lapisan ini terbagi menjadi dua sub lapisan yaitu *Logical Link Control* (LLC) dan *Media Access Control* (MAC). Kedua lapisan ini memiliki fungsi untuk memindahkan paket *menuju* dan keluar dari jaringan, Pada lapisan ini *bit* dan *byte* bergabung menjadi *frame*, atau sebaliknya. Sub lapisan LLC memaketkan *byte* yang diterima dari sublapisan MAC yang ada dibawah sehingga menjadi format yang mudah dibaca oleh lapisan jaringan diatasnya.

3. Lapisan Jaringan (*Network Layer*)

Lapisan jaringan adalah lapisan ketiga pada model OSI yang memiliki fungsi untuk melakukan rutinitas paket melalui multiple jaringan. Lapisan ini bekerja tanpa memperhatikan protokol pokok yang digunakan. Hal ini menyebabkan alat seperti *router* dapat beroperasi pada level ini dapat digunakan untuk menghubungkan jaringan – jaringan pada lapisan *datalink* dan lapisan fisik. Contoh : Manajemen *bandwidth* dengan disiplin antrian *Per Connection Queue* (PCQ) dan *Hierarchical Token Bucket* (HTB)

4. Lapisan *Transport* (*Transport Layer*)

Lapisan *transport* merupakan lapisan keempat pada model OSI. Lapisan ini berfungsi untuk mentransmisikan pesan dari *host* pengirim ke penerima. Lapisan *transport* bertugas membuat sirkuit *virtual* diantara dua titik didalam jaringan dan memastikan integritas *data* (jika *level* atau protokol dibawahnya tidak

menyediakan *service* ini). Contoh : *Transmission Control Protocol* (TCP), *Name Binding Protocol* (NBP) dan *User Datagram Protocol* (UDP).

5. Lapisan Sesi (*Session Layer*)

Lapisan kelima ini memiliki fungsi untuk mengadakan, mempertahankan dan memutuskan komunikasi diantara aplikasi-aplikasi atau proses-proses yang berjalan di jaringan. Contoh : *SQL*, *Apple Talk Session Protocol* (ASP), dan *Digital Network Architecture Session Control Program* (DNASCP).

6. Lapisan Presentasi (*Presentation Layer*)

Lapisan presentasi merupakan lapisan keenam pada model OSI. Lapisan ini sangat erat kaitannya dengan lapisan aplikasi. Tugas utamanya adalah untuk memastikan bahwa *data* yang sedang dilewatkan menuju lapisan aplikasi sudah dikonversi menjadi *format* yang dimengerti lapisan aplikasi. Contoh : *American Standard Code for Information Interchange* (ASCII), *Joint Photographic Experts Group* (JPEG), dan *Musical Instrument Digital Interface* (MIDI).

7. Lapisan Aplikasi (*Application Layer*)

Lapisan ini merupakan lapisan teratas pada model OSI. Lapisan aplikasi digunakan untuk menyediakan akses jaringan untuk program aplikasi. Program aplikasi pengguna dan *service* sistem biasanya memperoleh akses jaringan melalui interaksi dengan proses yang sedang berjalan pada lapisan OSI ini. Contoh : *Email* (pop3 dan SMTP), *File Transfer Protocol* (FTP), dan *Hypertext Transfer Protocol* (HTTP).[8]

2.6 Model Referensi *Transmission Control Protocol/Internet Protocol* (TCP/IP)

Transmission Control Protocol/Internet Protocol (TCP/IP) adalah satu set aturan standar komunikasi *data* yang digunakan dalam proses transfer *data* dari satu komputer ke komputer lain di jaringan komputer tanpa melihat perbedaan jenis *hardware*. Protokol TCP/IP dikembangkan dalam riset pertama kali oleh *Defense Advanced Research Projects Agency* (DARPA) di Amerika Serikat dan paling banyak digunakan saat ini yang implementasinya dalam bentuk perangkat lunak (*software*) di sistem operasi.

2.6.1 Karakteristik Lapisan TCP/IP

TCP/IP mengatur komunikasi *data* komputer di *internet* dan memastikan pengiriman *data* ke alamat yang dituju. Lapisan-lapisan protokol TCP/IP melayani permintaan pengguna untuk mengirim dan menerima *data*, mengatur komunikasi antar-*host*, melakukan pengecekan kesalahan, menyampaikan paket ke alamat yang benar, dan mengirim/menerima data dari media fisik. protokol TCP/IP mempunyai empat lapisan yang merupakan penyederhanaan dari lapisan

2.6.2 Lapisan-Lapisan Model TCP/IP

Protokol TCP/IP terdiri atas protokol-protokol yang melakukan fungsi masing-masing secara spesifik. Protokol-protokol ini dikategorikan berdasarkan fungsi dalam lapisan-lapisan tertentu. TCP/IP memiliki empat lapisan kategori untuk protokol-protokol yang ada. Lapisan-lapisan tersebut adalah :

1. Lapisan Aplikasi (*Application Layer*)

Lapisan Aplikasi pada model protokol TCP/Ip adalah lapisan yang melayani permintaan pengguna untuk mengirim dan menerima *data*. Lapisan ini merupakan tempat di mana aplikasi dan servis-servis lain memperoleh akses ke jaringan. Lapisan ini merupakan “jendela” protokol terhadap dunia. Dua API (*Application Programming Interface/Antarmuka Program Aplikasi*) yang berbeda menyediakan akses ke protokol transportasi TCP/IP, yaitu *Socket Window* dan *Net Bios*.

2. Lapisan Transportasi (*Transport Layer*)

Lapisan Transportasi pada model protokol TCP/IP berfungsi mengatur komunikasi antar *host* dan melakukan pengecekan kesalahan. Lapisan ini melakukan dan mempertahankan komunikasi *point-to-point* di antara dua *host*. Fungsi utamanya adalah memberi balasan terhadap informasi yang diterima, mengontrol aliran, mengurutkan dan mentransmisikan paket-paket *data*.

Jenis service yang diperlukan tergantung kepada aplikasi, dapat menggunakan *Transmission Control Protocol* (TCP) atau *User Datagram Protocol* (UDP). TCP biasanya digunakan ketika aplikasi yang dijalankan memerlukan transmisi *data* dalam jumlah besar dan memastikan *data* yang diperoleh penerima merupakan *data* yang benar dan dalam waktu pengiriman yang tepat. Lapisan *Internet* (*Internet Layer*)

Lapisan *internet* bertugas untuk menjalankan *data* di dalam dan di antara jaringan yang berbeda. *Router* sangat berfungsi pada lapisan protokol ini dan bertugas meneruskan paket *data* dari suatu jaringan atau segmen kepada jaringan lainnya.

3. Lapisan Antarmuka Jaringan/Fisik (*Network/Physical Interface*)

Lapisan antarmuka jaringan merupakan lapisan paling bawah. Lapisan ini bertugas mengirimkan paket *data* yang berisi IP. Lapisan ini bekerja dekat dengan *Address Resolution Protocol* (ARP) untuk menentukan *header* informasi yang tepat yang perlu ditambahkan pada masing-masing *frame*. Lapisan ini membuat *frame* yang cocok dengan tipe jaringan yang digunakan, seperti *Ethernet*, *Token Ring*, atau *Automated Teller Machine* (ATM), kemudian meletakkan paket *data* IP ke daerah muatan pada sebuah *frame* dan mengirimkan *data*. [8]

2.7 Layanan Jaringan

Layanan suatu jaringan sangat bermacam- macam jenisnya. Layanan ini didapat dengan cara mengakses suatu *server* yang ada didalam suatu *server* yang ada pada suatu jaringan. Bentuk dari layanan jaringan tersebut bermacam – macam tergantung dari *server* yang diakses.

2.7.1 Hypertext Transfer Protocol (HTTP)

Hypertext Transfer Protocol (HTTP) adalah sebuah protokol jaringan pada lapisan aplikasi yang digunakan untuk sistem informasi terdistribusi, kolaboratif dan menggunakan *hypermedia*. HTTP juga merupakan sebuah protokol meminta dan menjawab antara *client* dan *server*. Penggunaannya banyak pada pengambilan sumber daya yang saling terhubung dengan tautan, yang disebut dengan dokumen *hypertext*, yang kemudian membentuk *World Wide Web* (WWW) pada tahun 1990 oleh fisikawan Inggris Tim Berners-Lee.

Hingga kini, ada dua versi mayor dari protokol http, yakni http/1.0 yang menggunakan koneksi terpisah untuk setiap dokumen, dan http/1.1 yang dapat menggunakan koneksi yang sama untuk melakukan transaksi. Dengan demikian, http/1.1 bisa lebih cepat karena memang tidak usah membuang waktu untuk pembuatan koneksi berulang-ulang. *Client* yang mengirimkan permintaan http disebut dengan *user agent*.

2.7.2 File Transfer Protocol (FTP)

File Transfer Protocol (FTP) merupakan protokol yang bertanggung jawab pada pertukaran *file* di *internet*. FTP bekerja seperti layaknya protokol HTTP dalam menyajikan halaman web. Umumnya, *FTP server* merupakan suatu komputer yang khusus menyimpan *file – file* yang dapat diunduh dari *internet*.

Model Protokol FTP adalah sebagai berikut :

1. Control Connection

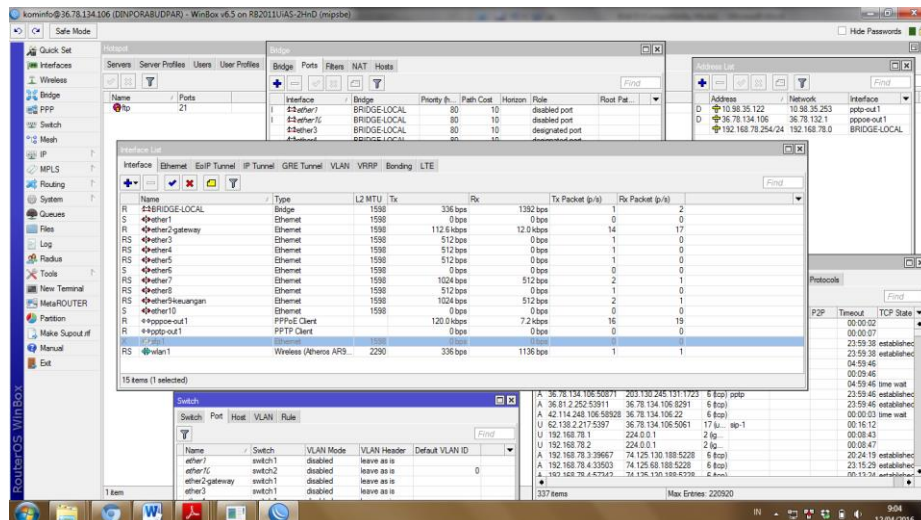
Control Connection digunakan untuk hubungan antara *server* dan *client*. *Server* membuka diri secara pasif di sebuah *port* khusus yaitu *port 21*.

2. Data Connection

Data Connection yang dibangun setiap kali sebuah file ditransfer dari *server* ke *client*. *Port* yang digunakan yaitu *port 20*. [7]

2.8 Winbox

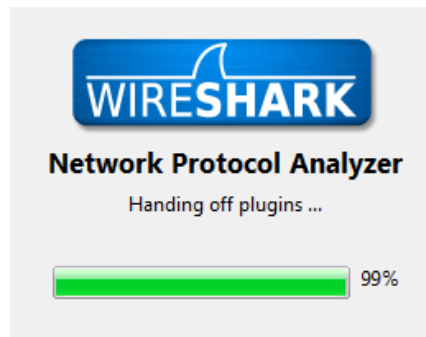
Winbox adalah sebuah *utility* yang digunakan untuk melakukan *remote* ke *server* mikrotik dalam mode *Graphical User Interface* (GUI). Sehingga dengan cara ini tidak perlu bersusah payah mengetikkan perintah *Command Line* (CLI) mikrotik. Mengkonfigurasi mikrotik melalui winbox ini lebih banyak digunakan karena selain penggunaannya yang mudah, juga tidak harus menghafalkan perintah-perintah pada mikrotik. Semua perintah itu sudah tersedia dalam bentuk *Graphical Menu* pada Winbox, sehingga *user* hanya perlu memilih menu yang diinginkan yang terdapat di winbox tanpa perlu mengetikkan perintah pada mikrotik. [9]



Gambar 2.4 Tampilan Winbox

2.9 Wireshark

Wireshark adalah *tool* yang ditujukan untuk penganalisan paket *data* jaringan. Penganalisaan Kinerja Jaringan itu dapat melingkupi berbagai hal, mulai dari proses menangkap paket-paket *data* atau informasi yang berlalu-lalang dalam jaringan, sampai digunakan pula untuk *sniffing* (memperoleh informasi penting seperti *password email*, dll). Wireshark bisa digunakan secara gratis karena aplikasi ini berbasis sumber terbuka. Aplikasi Wireshark dapat berjalan dibanyak *platform*, seperti Linux, Windows, dan Mac. Wireshark melakukan pengawasan paket secara waktu nyata (*real time*) yang kemudian menangkap *data* dan menampilkannya selengkap mungkin.



Gambar 2.5 Wireshark *Network Protocol Analyzer*

Wireshark dapat dikatakan sebagai *tool* analisis paket *data* jaringan yang paling sering digunakan. Berikut adalah fitur pada Wireshark :

1. Tersedia untuk *platform* UNIC, Linux, Windows, dan Mac.
2. Dapat melakukan *capture* paket *data* jaringan secara *real time*.
3. Dapat menampilkan informasi protokol secara lengkap.
4. Paket *data* dapat disimpan menjadi *file* dan nantinya dapat dibuka kembali.
5. Pemfilteran paket jaringan
6. Pencarian paket *data* dengan kriteria spesifik.
7. Pewarnaan penampilan paket *data* sehingga mempermudah penganalisan paket *data*.
8. Menampilkan *data* statistik. [10]