

BAB II

TEORI PENUNJANG

2.1 Kejaksaan Republik Indonesia

Kejaksaan Republik Indonesia adalah lembaga negara yang melaksanakan kekuasaan negara, khususnya dibidang penuntutan. Sebagai badan yang berwenang dalam penegakan hukum dan keadilan, Kejaksaan dipimpin oleh Jaksa Agung yang dipilih dan bertanggung jawab kepada Presiden. Kejaksaan Agung, Kejaksaan Tinggi, dan Kejaksaan Negeri merupakan kekuasaan negara khususnya dibidang penuntutan, dimana semuanya merupakan satu kesatuan yang utuh yang tidak dapat dipisahkan.

Mengacu pada Undang-Undang No. 16 Tahun 2004 yang menggantikan Undang-Undang No. 5 Tahun 1991 tentang Kejaksaan Republik Indonesia, Kejaksaan sebagai salah satu lembaga penegak hukum dituntut untuk lebih berperan dalam menegakkan supremasi hukum, perlindungan kepentingan umum, penegakan hak asasi manusia, serta pemberantasan Korupsi, Kolusi, dan Nepotisme (KKN). Didalam Undang-Undang Kejaksaan yang baru ini, Kejaksaan Republik Indonesia sebagai lembaga negara yang melaksanakan kekuasaan negara dibidang penuntutan harus melaksanakan fungsi, tugas, dan wewenangnya secara merdeka, terlepas dari pengaruh kekuasaan pemerintah dan pengaruh kekuasaan lainnya (Pasal 2 ayat 2 Undang-Undang Nomor 16 Tahun 2004).

Dalam menjalankan tugas dan wewenangnya, Kejaksaan dipimpin oleh Jaksa Agung yang membawahi enam Jaksa Agung Muda serta 31 Kepala Kejaksaan Tinggi pada tiap provinsi. Undang-Undang No. 16 Tahun 2004 tentang Kejaksaan Republik Indonesia juga mengisyaratkan bahwa lembaga Kejaksaan berada pada posisi sentral dengan peran strategis dalam pemantapan ketahanan bangsa, karena Kejaksaan berada di poros dan menjadi filter antara proses penyidikan dan proses pemeriksaan di persidangan serta juga sebagai pelaksana penetapan dan keputusan pengadilan. Sehingga, Lembaga Kejaksaan sebagai

pengendali proses perkara (*Dominus Litis*), karena hanya institusi Kejaksaan yang dapat menentukan apakah suatu kasus dapat diajukan ke Pengadilan atau tidak berdasarkan alat bukti yang sah menurut Hukum Acara Pidana.

Perlu ditambahkan, Kejaksaan juga merupakan satu-satunya Instansi pelaksana putusan pidana (*executive ambtenaar*). Selain berperan dalam perkara pidana, Kejaksaan juga memiliki peran lain dalam Hukum Perdata dan Tata Usaha Negara, yaitu dapat mewakili Pemerintah dalam Perkara Perdata dan Tata Usaha Negara sebagai Jaksa Pengacara Negara. Jaksa sebagai pelaksana kewenangan tersebut diberi wewenang sebagai Penuntut Umum serta melaksanakan putusan pengadilan, dan wewenang lain berdasarkan Undang-Undang.[1]

2.2 Pengertian Jaringan Komputer

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 teks, gambar, video, dan suara. Dinamakan sebagai jaringan komputer jika didalam sekumpulan komputer tersebut dihubungkan melalui media fisik dan perangkat lunak yang memfasilitasi komunikasi antara komputer-komputer tersebut.

Jaringan komputer merupakan sekumpulan komputer berjumlah banyak yang terpisah-pisah akan tetapi saling berhubungan dalam melaksanakan tugasnya. Dua buah komputer misalnya dikatakan terkoneksi bila keduanya dapat saling bertukar informasi. Bentuk koneksi dapat melalui: kabel tembaga, serat optic, gelombang mikro, dan satelit komunikasi.[2]

Jaringan komputer dapat dibagi berdasarkan beberapa klasifikasi, diantaranya:

1. Berdasarkan area geografis
2. Berdasarkan fungsi
3. Berdasarkan media transmisi

Berikut adalah penjelasan dari beberapa klasifikasi tersebut:

2.2.1 Berdasarkan Area Geografis

Jaringan komputer dapat dibedakan berdasarkan cakupan geografisnya. Ada tiga kategori utama jaringan komputer yaitu:[2]

1. LAN (*Local Area Network*)

LAN merupakan jaringan komputer dengan ruang lingkup terbatas, meliputi lokasi seperti gedung, kampus, kantor. Sebuah LAN dapat dibangun minimal 2 komputer dengan spesifikasi rendah sekalipun. LAN akan menjadikan komputer terhubung dengan komputer lain, sehingga komputer tersebut bisa saling berinteraksi.

2. MAN (*Metropolitan Area Network*)

MAN adalah suatu jaringan dalam suatu kota dengan transfer data berkecepatan tinggi, yang menghubungkan berbagai lokasi seperti kampus, perkantoran dan sebagainya. Jaringan MAN adalah gabungan dari beberapa LAN. Jangkauan dari MAN ini antara 10 – 50 km. MAN ini merupakan jaringan yang tepat untuk membangun jaringan antar kantor-kantor dalam satu kota antara instansi dan kantor pusat yang berada dalam jangkauannya.

3. WAN (*Wide Area Network*)

WAN merupakan jaringan yang mencakup area yang lebih luas sebagai contoh adalah jaringan komputer antar wilayah, antar negara, maupun antar benua, atau dapat didefinisikan juga sebagai jaringan komputer yang membutuhkan *router* dan saluran komunikasi publik. WAN digunakan untuk menghubungkan jaringan area lokal yang satu dengan jaringan lokal lainnya, sehingga pengguna atau komputer di lokasi yang satu dapat berkomunikasi dengan pengguna dan komputer di lokasi lainnya.

2.2.2 Berdasarkan Media Penghantar

Berdasarkan media transmisi, jaringan komputer dapat dibagi menjadi dua jenis, yaitu:[3]

1. Jaringan Berkabel (*Wire Network*)

Wire Network adalah jaringan komputer yang menggunakan kabel sebagai media penghantar. Kabel jaringan berfungsi dalam mengirim informasi dalam bentuk sinyal listrik antar komputer. Jadi, data mengalir pada kabel. Kabel yang digunakan adalah kabel *twisted pair*, kabel *coaxial* dan kabel serat optik. Media transmisi ini cocok digunakan untuk LAN, MAN dan WAN. Untuk LAN biasanya menggunakan kabel UTP (*unshielded twisted pair*), sedangkan untuk MAN dan WAN menggunakan serat optik.

2. Jaringan Nirkabel (*Wireless Network*)

Wireless Network adalah jaringan tanpa kabel yang menggunakan gelombang radio sebagai media penghantarnya biasa disebut dengan *wireless*. Satelit telekomunikasi, *Bluetooth*, dan *wifi* adalah contoh dari media transmisi *wireless*.

2.2.3 Berdasarkan Fungsi

Berdasarkan fungsinya, jaringan komputer dapat dibagi menjadi dua jenis, yaitu:

1. *Client Server*

Client Server adalah jaringan komputer yang salah satu komputer difungsikan sebagai server atau induk dari komputer lain. Server melayani komputer lain yang disebut *client*. Layanan yang diberikan bisa berupa akses *Web*, *e-mail*, *file*, *database* dan lain-lain. Sedangkan komputer *client* menerima pelayanan dari komputer server.

2. *Peer To Peer*

Pada jaringan ini masing-masing komputer dapat berfungsi sebagai *server* dan *client* yang berarti setiap komputer dapat memberikan layanan dan dapat menerima dari komputer atau ke komputer lain.

2.3 Topologi Fisik (*Physical Topology*)

Topologi fisik adalah sebuah aturan atau cara untuk menghubungkan setiap perangkat agar dapat saling berkomunikasi dengan membentuk jalur komunikasi menggunakan media transmisi. Ada empat topologi dasar diantaranya:

1. Topologi *Bus*

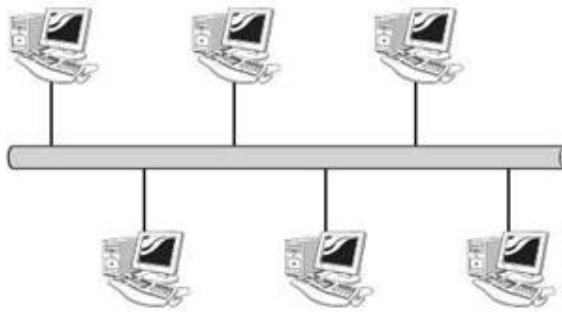
Topologi *bus* hanya menyediakan satu saluran untuk komunikasi ke semua perangkat sehingga setiap perangkat harus bergantian menggunakan saluran tersebut. Dengan kata lain hanya dua perangkat yang bisa berkomunikasi, bila perangkat lain ingin berkomunikasi perangkat itu harus menunggu gilirannya.

Kelebihan dari topologi *bus* adalah:

- Mudah dan sederhana untuk menambahkan komputer ke jaringan lain, hanya perlu memasang konektor baru.
- Cocok untuk jaringan komputer kecil karena mudah di-*setup*.
- Jumlah terminal dapat ditambah atau dikurangi tanpa mengganggu kerja komputer lain.
- Biaya pembuatannya lebih murah dibandingkan dengan topologi lain.

Kekurangan dari topologi *bus* adalah:

- Harus ada terminasi di ujung-ujung *bus*.
- Kesulitan perawatan jika dalam jumlah besar.
- Sering terjadi tabrakan data.
- Seluruh jaringan akan mati jika ada kerusakan pada kabel utama.



Gambar 2.1 Topologi *Bus*

2. Topologi *Star*

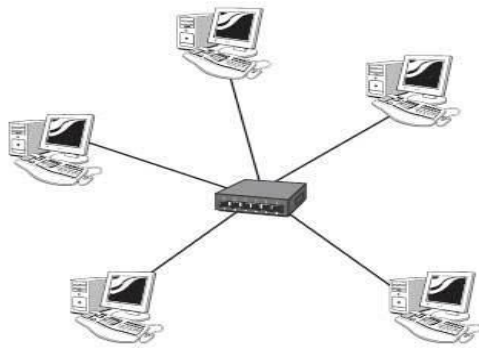
Topologi model ini didesain dimana setiap node (*file server*, *workstation*, dan perangkat lainnya) terkoneksi ke jaringan melewati sebuah *hub* atau *switch*. Data yang dikirim jaringan akan melewati *hub* sebelum melanjutkan ke tempat tujuan. *Hub* akan mengatur keseluruhan fungsi jaringan yang bertindak sebagai penguat aliran data. Konfigurasi pada jaringan model ini menggunakan kabel koaksial atau kabel serat optik.

Kelebihan dari topologi *star* adalah:

- Akses kontrol terpusat.
- Bila terjadi kerusakan pada saluran salah satu perangkat tidak mempengaruhi perangkat lain.
- Kemudahan pengelolaan jaringan sehingga cocok digunakan untuk jumlah perangkat yang banyak.
- Tahan terhadap lalu lintas jaringan yang sibuk.

Kekurangan dari topologi *star* adalah:

- Jika *hub* atau *switch* rusak, semua jaringan rusak.
- Jaringan tergantung kepada *hub* atau *switch*
- *Hub* atau *switch* jadi elemen kritis karena kontrol terpusat.
- Boros kabel, dan mahal biaya.



Gambar 2.2 Topologi *Star*

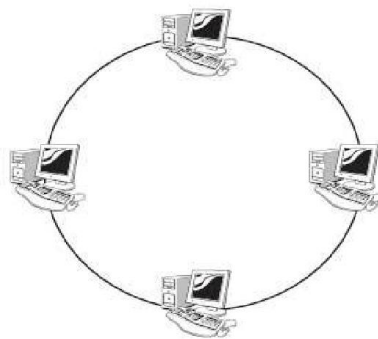
3. Topologi *Ring*

Topologi jaringan yang berupa lingkaran tertutup yang berisi *node-node*. Sinyal mengalir dalam dua arah sehingga dapat menghindarkan terjadinya *collision*, sehingga memungkinkan terjadinya pergerakan data yang sangat cepat.

- Tidak akan terjadi tabrakan data karena pada satu waktu hanya satu perangkat yang bisa mengirimkan data.
- Dapat melayani lalu lintas pengiriman data yang padat.

Kekurangan dari topologi *ring* adalah:

- Jika terjadi gangguan di satu titik, maka akan berpengaruh pada seluruh komputer.
- Sulitnya dalam penambahan dan pengurangan komputer.



Gambar 2.3 Topologi *Ring*

4. Topologi *Mesh*

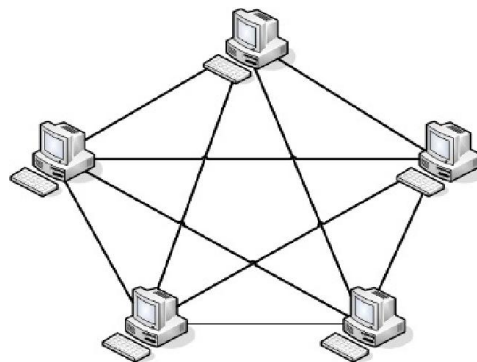
Topologi *mesh* adalah suatu bentuk hubungan antar perangkat dimana setiap perangkat terhubung secara langsung ke perangkat yang lainnya yang ada didalam jaringan. Oleh karena itu di topologi *mesh* ini setiap perangkat dapat langsung saling berhubungan.

Kelebihan dari topologi *mesh* adalah:

- Dapat mengirimkan data secara bersamaan ke setiap perangkat.
- Tingkat keamanan data tinggi.

Kekurangan topologi *mesh* adalah:

- Membutuhkan banyak kabel.
- Instalasi dan konfigurasi yang sangat rumit.



Gambar 2.4 Topologi *Mesh*

5. Topologi *Tree*

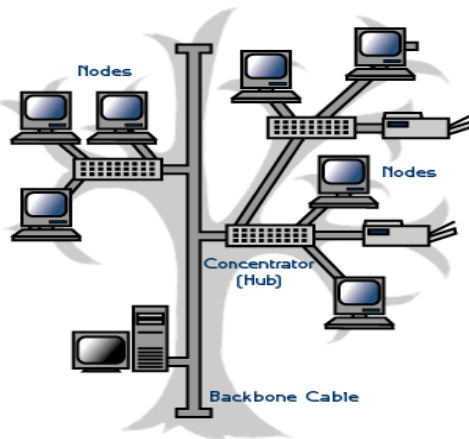
Topologi *tree* merupakan perpaduan antara topologi *bus* dan *star*, yang terdiri dari kelompok-kelompok dari *workstation* konfigurasi topologi *star* yang terkoneksi ke kabel utama yang menggunakan topologi *bus*. Topologi ini memungkinkan untuk pengembangan jaringan yang telah ada dan memungkinkan sebuah perusahaan melakukan konfigurasi jaringan sesuai dengan kebutuhannya

Kelebihan dari topologi *tree* adalah:

- Sangat baik untuk jaringan komputer yang besar.
- Mengatasi kekurangan dari topologi *star*.
- Instalasi jaringan dari titik ke titik pada masing-masing segmen.

Kekurangan dari topologi *tree* adalah:

- Jika jaringan utama rusak, seluruh segmen akan *down*.
- Sangat sulit untuk dikonfigurasi dan juga untuk pengkabelnya dibandingkan topologi jaringan model lain.
- Memerlukan kabel yang sangat banyak sehingga diperlukan perencanaan yang matang.



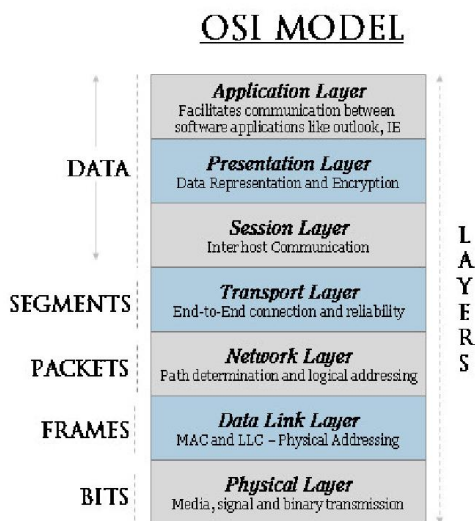
Gambar 2.5 Topologi Pohon (*Tree Topology*)

2.4 Protokol Jaringan Komputer

Dalam jaringan komputer, terjadi komunikasi diantara sekumpulan *entity* yang memiliki sistem berbeda. *Entity* adalah segala sesuatu yang dapat mengirim atau menerima informasi. Karena memiliki sistem yang berbeda, *entity* tidak dapat begitu saja berkomunikasi dengan *entity* lainnya. Maka dibutuhkan protokol yaitu sebuah aturan atau standar yang mengatur atau mengijinkan terjadinya hubungan, komunikasi dan perpindahan data antara dua atau lebih komputer didalam sebuah jaringan. Protokol mendefinisikan apa yang dikomunikasikan, bagaimana dan kapan terjadinya komunikasi.

2.4.1 OSI (*Open System Interconnection*) Model

OSI (*Open System Interconnection*) model adalah standard atau kerangka kerja yang berlapis dirancang untuk sitem jaringan yang memungkinkan semua jenis sistem komputer yang ada di dunia dapat berkomunikasi. Tujuan utama penggunaan OSI model adalah untuk membantu desainer jaringan memahami fungsi dari tiap-tiap *layer* yang berhubungan dengan aliran komunikasi data, termasuk jenis-jenis protokol jaringan dan metode transmisi. Model OSI dibagi menjadi 7 lapisan, dengan karakteristik dan fungsinya masing-masing. Tiap lapisan harus dapat berkomunikasi dengan lapisan yang berada di atasnya maupun di bawahnya secara langsung melalui beberapa protokol dan standar.



Gambar 2.6 Model OSI

Berikut penjelasan dari setiap lapisan yang memiliki fungsi masing-masing pada OSI Model:

1. *Physical layer*

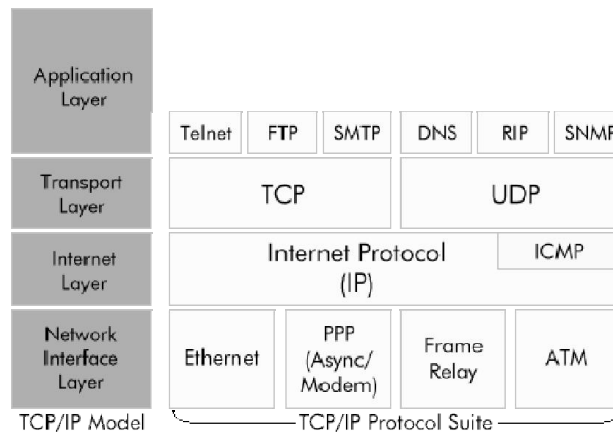
Lapisan fisik megkoordinasikan fungsi yang diperlukan untuk membawa aliran bit melalui media fisik. Hal tersebut berkaitan dengan spesifikasi mekanik dan listrik dari perangkat atau media transmisi seperti besar tegangan dan jenis kabel.

2. *Data Link layer*
Pada lapisan ini bertugas membagi aliran bit yang diterima dari lapisan *network* menjadi unit-unit data yang dikendalikan yang disebut *frame*. Selain itu, pada *level* ini terjadi koreksi kesalahan, *flow control*, pengalamatan perangkat keras seperti halnya *Media Acces Control* (MAC address) dan menentukan bagaimana perangkat-perangkat jaringan seperti *hub*, *bridge* dan *repeater* beroperasi.
3. *Network layer*
Lapisan ini menentukan rute pengiriman dan mendefinisikan alamat-alamat IP, membuat *header* untuk paket-paket, dan kemudian melakukan *routing* dan *switch layer 3* agar data sampai ditempat tujuan dengan benar.
4. *Transport layer*
Lapisan ini mengatur keutuhan data, menerima data dari lapisan *session* dan meneruskan ke lapisan *network*. Lapisan ini berfungsi untuk memecah data kedalam paket-paket data serta memberikan nomor urut ke paket-paket tersebut sehingga dapat disusun kembali pada sisi tujuan setelah diterima.
5. *Session layer*
Lapisan ini menyiapkan saluran komunikasi dan terminal dalam hubungan antar terminal, mengkoordinasikan proses pengiriman dan penerimaan serta mengatur pertukaran data.
6. *Presentation layer*
Pada lapisan ini dilakukan konversi data agar data yang dikirim dapat dimengerti oleh penerima, kompresi teks dan penyediaan data.
7. *Application layer*
Lapisan paling atas ini mengatur interaksi pengguna komputer dengan program aplikasi yang dipakai. Lapisan ini juga mengatur bagaimana aplikasi dapat mengakses jaringan dan kemudian membuat pesan-pesan kesalahan. Protocol yang berada pada lapisan ini adalah HTTP, FTP, SMTP dan NFS.

Ketika data ditransfer melalui jaringan, maka data sebelumnya harus melewati ketujuh *layer*, mulai dari *application layer* sampai *physical layer*, kemudian di sisi penerima data tersebut juga melewati *layer physical* sampai *application layer*. Pada saat data melewati satu persatu *layer* dari sisi pengirim, maka akan ditambahkan satu *header* sedangkan pada sisi penerima *header* dibuka sesuai dengan *layer*-nya.

2.4.2 TCP/IP Protocol Suite

TCP/IP *protocolsuite* dikembangkan sebelum model OSI. Oleh karena itu, lapisan di TCP/IP *protocolsuite* tidak sama persis dalam model OSI. TCP/IP *protocolsuite* memiliki empat lapisan yaitu *network*, *Internet*, *transport*, dan *application*. Namun, ketika TCP / IP dibandingkan dengan OSI dapat dikatakan bahwa lapisan *network* setara dengan gabungan dari *layer data link* dan *physical*. *Internetlayer* setara dengan *layernetwork*, dan *applicaton layer* mengerjakan pekerjaan yang sama seperti *session*, *presentation*, dan *application layer*. [2]



Gambar 2.7 Protokol TCP/IP

1. *Application Layer* merupakan lapisan dalam arsitektur TCP/IP yang berfungsi mendefinisikan aplikasi-aplikasi yang dijalankan pada jaringan.

Karena itu terdapat banyak protokol pada lapisan ini, sesuai dengan banyaknya aplikasi TCP/IP yang dapat dijalankan. Contohnya adalah SMTP (*Simple Mail Transfer Protocol*) untuk pengiriman *e-mail*, dan FTP (*File Transfer Protocol*) untuk transfer file

2. *Transport Layer* mendefinisikan cara untuk melakukan pengiriman data antara *end to end* secara handal. Lapisan ini menjamin bahwa informasi yang diterima pada sisi penerima adalah sama dengan informasi yang dikirimkan pada pengirim.
3. *Internet Layer* mendefinisikan bagaimana hubungan dapat terjadi antara dua pihak yang berada pada jaringan yang berbeda seperti lapisan *network* pada OSI. Pada jaringan *Internet* yang terdiri atas puluhan juta *host* dan ratusan ribu jaringan lokal, lapisan ini bertugas untuk menjamin agar suatu paket yang dikirimkan dapat menemukan tujuannya dimana pun berada. Oleh karena itu, lapisan ini memiliki peranan penting terutama dalam mewujudkan *interworking* yang meliputi wilayah luas (*worldwide Internet*).
4. *Network Layer* berfungsi yang mirip dengan lapisan Data-Link pada OSI. Lapisan ini mengatur penyaluran data *frame* data pada media fisik yang digunakan secara handal. Lapisan ini biasanya memberikan servis untuk deteksi dan koreksi kesalahan dari data yang ditransmisikan. Beberapa contoh protokol yang digunakan pada lapisan ini adalah X.25 jaringan public, *Ethernet* untuk jaringan *Ethernet*, AX.25 untuk jaringan paket radio.

2.5 IP (*Internet Protocol*) Address

Sebuah komputer tentu perlu untuk saling bertukar informasi dengan komputer lain pada jaringan yang berbeda. Biasanya komputer berkomunikasi menggunakan Internet. Paket yang ditransmisikan akan melewati beberapa jaringan LAN atau WAN sebelum mencapai komputer tujuan. *IP Address* adalah

sebuah alamat logika pada sebuah perangkat dalam jaringan (komputer atau *router*) yang berada dalam *network layer* pada protokol TCP/IP.

2.5.1 Pengalamatan IPv4

Pengalamatan IPv4 adalah sebuah jenis pengalamatan jaringan yang digunakan di dalam protokol jaringan TCP/IP yang terdiri dari sekumpulan bilangan biner sepanjang 32bit, yang dibagi menjadi 4 segmen dan setiap segmen terdiri dari 8bit. IP *address* merupakan identifikasi setiap *host* pada jaringan Internet.

Untuk memudahkan pengaturan IP *address*, dibentuklah suatu badan yang mengatur pembagian IP *address*. Badan tersebut bernama InterNIC (*Internet Network Information Center*). InterNIC membagi IP *address* menjadi beberapa kelas. Kelas-kelas tersebut meliputi:

1. IP *Address* kelas A

Jika bit pertama dari IP *address* adalah 0, maka IP *address* tersebut termasuk kedalam *network* kelas A. bit awal dan 7bit berikutnya (8bit pertama) merupakan bit untuk *network id*, sedangkan 24bit terakhir merupakan bit untuk *host id*. Dengan demikian, hanya ada 128 *network id* kelas A, yaitu dari 0xx.xxx.xxx sampai 127.xxx.xxx.xxx. Setiap *network id* dapat menampung lebih dari 16juta *host*.(xxx adalah variabel, bernilai 0 sampai dengan 255). Ilustrasi dapat dilihat pada tabel 2.1 berikut:

Tabel 2.1 IP *Address* kelas A

0 – 127	0 – 255	0 – 255	0 – 255
0nnnnnnn	Hhhhhhhh	Hhhhhhhh	hhhhhhhh
Bit Network ID	Bit Host ID		

2. IP *Address* kelas B

Jika 2bit pertama dari IP *address* adalah 10, maka IP *address* tersebut termasuk kedalam *network* kelas B. Dua bit awal dan 14bit berikutnya (16bit pertama) merupakan bit *network id*, sedangkan 16bit terakhir merupakan bit *host id*. Dengan demikian, terdapat lebih dari 16ribu *network id* kelas B, yaitu dari 128.0xx.xxx.xxx sampai 191.255.xxx.xxx. Setiap *network id* dapat menampung lebih dari 65ribu *host*. (xxx adalah variabel, bernilai 0 sampai dengan 255). Ilustrasi dapat dilihat pada tabel 2.2 berikut:

Tabel 2.2 IP *Address* kelas B

128 – 191	0 – 255	0 – 255	0 – 255
10nnnnnn	Nnnnnnnn	Hhhhhhhh	hhhhhhhh
Bit Network ID		Bit Host ID	

3. IP *Address* kelas C

Jika 3bit pertama dari IP *address* adalah 110, maka IP *address* tersebut termasuk kedalam *network* kelas C. tiga bit awal dan 21bit berikutnya (24bit pertama) merupakan bit *network id*, sedangkan 8bit terakhir merupakan bit *host id*. Dengan demikian, terdapat lebih dari 2juta *network id* kelas C, yaitu dari 192.0.0.xxx sampai 223.255.255.xxx. Setiap *network* kelas C hanya mampu menampung sekitar 256 *host*. Ilustrasi dapat dilihat pada tabel 2.3 berikut:

Tabel 2.3 IP *Address* kelas C

192 – 255	0 – 255	0 – 255	0 – 255
110nnnnn	Nnnnnnnn	Nnnnnnnn	hhhhhhhh
Bit Network ID			Bit Host ID

2.6 Perangkat Jaringan Komputer

Pada jaringan komputer terdapat komponen dan fitur yang mendukung agar jaringan komputer dapat berjalan dengan lancar dan memudahkan pengaturannya. Berikut adalah beberapa jenis perangkat jaringan komputer yang digunakan:

2.6.1 Kabel UTP (*Unshilded Twisted Pair*)

UTP (*Unshilded Twisted Pair*) merupakan salah satu jenis kabel yang paling banyak digunakan dalam jaringan komputer. Kabel ini berisi empat pasang (*Pair*) kabel yang setiap pasangannya dipilin (*Twisted*). Kabel ini tidak dilengkapi dengan pelindung (*Unshilded*). EIA/TIA (*Electronic Industry Association / Telecommunication Industry Association*) telah menetapkan standar UTP dan lima kategori kecepatan kabel. Berikut merupakan tabel standar untuk kabel UTP:

Tabel 2.4Kategori standar EIA/TIA kabel UTP

Tipe	Penggunaan
Kategori 1	Hanya suara (kabel telepon)
Kategori 2	Data hingga 4Mbps (<i>LocalTalk</i>)
Kategori 3	Data hingga 10Mbps (<i>Ethernet</i>)
Kategori 4	Data hingga 20Mbps (<i>16Mbps Token Ring</i>)
Kategori 5	Data hingga 100Mbps (<i>Fast Ethernet</i>)

Kabel UTP ini juga dibagi menjadi dua jenis pemasangan yaitu *straight cable* dan *crossover cable*, dimana untuk kedua jenis ini memiliki perbedaan

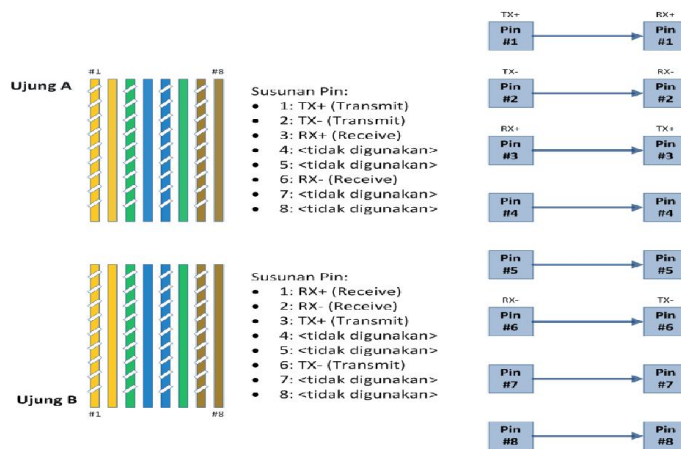
pemasangannya. Untuk *straight cable* kedua ujungnya memiliki pin-pin yang sama, sedangkan untuk *crossover cable* ujungnya memiliki pin-pin yang berbeda. Dibawah ini gambar kedua jenis pemasangan kabel tersebut:

1. *Straight cable*

Jenis ini biasanya digunakan untuk menghubungkan beberapa unit komputer melalui perantara *hub* atau *switch* yang berfungsi sebagai konsentrator.

Penggunaan kabel UTP model *straight cable* pada jaringan lokal biasanya akan membentuk topologi *star* (bintang) atau *tree* (pohon) dengan *hub* atau *switch* sebagai pusatnya. Jika sebuah *hub* atau *switch* tidak berfungsi, maka seluruh komputer yang terhubung dengan *hub* atau *switch* tersebut tidak dapat saling berhubungan.

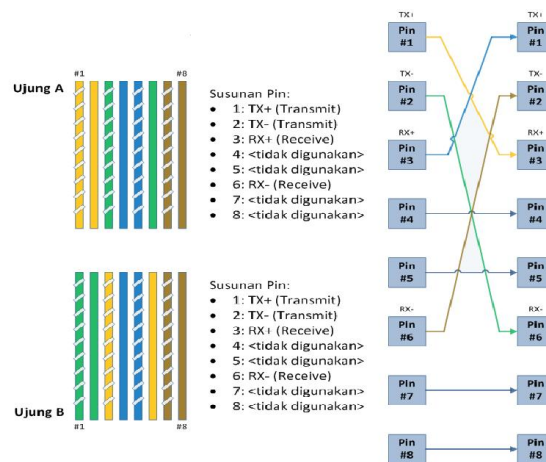
Penggunaan *hub* atau *switch* harus sesuai dengan kecepatan dari *ethernet card* yang digunakan pada masing-masing komputer. Karena perbedaan kecepatan pada NIC dan *hub* atau *switch* berarti kedua perangkat tersebut tidak dapat saling berkomunikasi secara maksimal.



Gambar 2.8 Susunan *Straight Cable*

2. *Crossover cable*

Berbeda dengan susunan *straight cable*, penggunaan kabel menyilang ini digunakan untuk berkomunikasi antar komputer (langsung tanpa *hub*), atau dapat juga digunakan untuk meng-*cascade* *hub* jika diperlukan. Sekarang ini ada beberapa jenis *hub* yang dapat di-*cascade* tanpa harus menggunakan kabel menyilang (*crossover*), tetapi juga dapat menggunakan *straight cable*.



Gambar 2.9 Susunan Crossover Cable

2.6.2 Network Interface Card (NIC)

NIC (*Network Interface Card*) atau *network adapter*. Merupakan sebuah *hardware* komputer yang didesain guna memungkinkan komputer berkomunikasi menggunakan jaringan dan juga menyediakan sistem *addressing low-level* melalui penggunaan *MAC address*. NIC memungkinkan *user* untuk terhubung satu dengan lainnya menggunakan kabel atau *wireless*. [3]

2.6.3 Switch

Switch mempunyai fungsi dan bentuk yang hampir sama dengan *HUB* yang digunakan sebagai pusat lalu lintas komunikasi data dalam topologi *star*.

Switch dapat mengenali alamat tujuan dari data, sehingga sinyal data yang dikirim tidak tersebar ke *port* atau komputer yang terkoneksi dalam satu jaringan.[2]

2.6.4 Router

Router memberikan kemampuan melakukan paket dari satu sistem ke sistem yang lain yang mungkin memiliki banyak jalur diantaranya. *Router* bekerja pada lapisan *Network* dalam model OSI. Umumnya *router* memiliki kecerdasan yang lebih tinggi daripada *bridge* dan dapat digunakan pada *internetwork* dengan tingkat kerumitan yang tinggi. *Router* yang saling terhubung dalam *internetwork* turut serta dalam sebuah algoritma terdistribusi untuk menentukan jalur optimum yang dilalui paket yang harus lewat dari suatu sistem ke sistem lain.[2]

2.7 Sistem Operasi Linux Ubuntu

Linux adalah nama yang diberikan kepada sistem operasi komputer bertipe Unix. Linux merupakan salah satu contoh hasil pengembangan perangkat lunak bebas dan *open source*. Terdapat banyak distribusi Linux atau yang lebih dikenal dengan Distro Linux yang berkembang sampai menghasilkan Distro turunan, contohnya adalah Distro Debian. Ubuntu adalah salah satu distribusi Linux yang berbasis pada Debian dan dirilis secara berkala (setiap enam bulan). Proyek Ubuntu disponsori oleh Canonical Ltd (perusahaan milik Mark Shuttleworth). Ubuntu adalah sistem operasi lengkap berbasis Linux, tersedia secara bebas dan mempunyai dukungan baik yang berasal dari komunitas maupun tenaga ahli profesional. Fokus utama sistem operasi Ubuntu adalah para pengguna dan kemudahan penggunaan dan pada setiap rilis Ubuntu akan memberikan perbaikan keamanan selama 18 bulan. Ubuntu menyertakan lingkungan desktop Gnome/KDE/XFCE terbaru di setiap rilis dan juga menyertakan beragam banyak pilihan perangkat lunak untuk server dan desktop yang semuanya bisa dikemas ke dalam satu CD. Komunitas Ubuntu dibentuk berdasarkan gagasan yang terdapat di dalam filosofi Ubuntu.

Kelebihan menggunakan Ubuntu, diantaranya:

1. Tidak perlu membeli lisensi dan boleh digunakan di banyak komputer hanya dengan satu CD.
2. Stabil, bebas virus, *malware*, *worm*, dan sebagainya sehingga tidak perlu memasang *antivirus*.
3. Sangat ringan dan boleh digunakan pada komputer dengan spesifikasi *hardware* yang rendah.
4. Tidak sulit untuk instalasi *driver* karena kebanyakan *driver* telah ada didalam CD seperti LAN, Wifi, Audio dan sebagainya.
5. Banyak aplikasi seperti untuk *browsing* Internet, *office*, kalkulator, dan sebagainya.
6. Cocok dipakai untuk sebuah server pada suatu jaringan komputer.

Kekurangan menggunakan Ubuntu, diantaranya:

1. Fitur Standar: setelah proses instalasi selesai, sistem operasi Ubuntu tidak sepenuhnya bisa dipakai langsung oleh pengguna. Dalam pemakaiannya ada beberapa fitur yang harus kita tambahkan sendiri setelah proses instalasi, seperti saat pengguna akan memutar mp3 file, untuk menjalankan file tersebut pengguna harus menambahkan *plugin* untuk mp3 tersebut. Namun jika ingin di instalasi secara *offline*, pengguna harus mempunyai CD/DVD Repository yang berisi segala kebutuhan yang terdapat didalamnya.
2. Koneksi Internet: dalam penginstalasian *software-software* tambahan dan *plugin*, Ubuntu sangat memerlukan koneksi Internet. Ubuntu yang telah kita instalasi terhubung dengan *mirror* untuk instalasi tambahan, *update* dan sebagainya. Pada saat kita akan menambahkan aplikasi kedalamnya melalui *Synaptic Package Manager*, atau instalasi manual melalui Terminal, nanti akan melewati *mirror site* yang terdaftar di OS tersebut, kemudian unduh aplikasi tersebut.
3. Tidak *User Friendly*: mungkin inilah salah satu kekurangan Ubuntu bagi para penggunanya. Dari segi penempatan menu, *management file* dan

properties, aplikasi-aplikasi didalamnya dan lainnya, akan membuat kaku para penggunanya karena ketidakbiasaan dalam menjalankannya. Dalam menggunakannta mungkin akan dibanding-bandingkan dengan OS yang sering mereka pakai sebelumnya. Dalam administrasinya seperti konfigurasi, instalasi dan sebagainya secara *default* Ubuntu masih menggunakan *file teks* dan memerlukan Terminal (*console*).[4]

2.8 *Chillispot*

Chillispot adalah *Wireless Access Point Controller* berbasis *open source*. *Chillispot* merupakan *software captive portal* yang digunakan untuk autentikasi *user Wireless Local Area Network (WLAN)*. Cara kerja *Chillispot* adalah dengan cara meng-*capture request* halaman *web client* dan kemudian di-*direct* ke halaman *web Chillispot* untuk *login* autentikasi. Data *user* dan *password* yang dimasukkan *user* akan ditransfer ke Server *RADIUS* untuk proses autentikasi dan otorisasi hak akses. Apabila data *user* dan *password* terautentikasi oleh Server *RADIUS* maka *user* dapat mengakses halaman *web* di Internet.

Chillispot dikembangkan pada *platform* sistem operasi *Linux* tetapi dapat juga di-*compile* pada sistem operasi *FreeBSD*, *OpenBSD*, *Solaris*, dan bahkan *MAC OS X*. *Chillispot* dikembangkan menggunakan bahasa pemograman *C* untuk meningkatkan portabilitas *platform* sistem operasi yang digunakan. Beberapa fitur yang dimiliki oleh *Chillispot* antara lain server *UAM*, layanan *DHCP*, dan *captive portal*.

Untuk membangun *hotspot* dengan autentikasi, *Chillispot* memerlukan beberapa komponen, yaitu:

1. Koneksi Internet
2. *Wireless LAN Access Point*
3. *Radius server*

2.8.1 CoovaChilli

CoovaChilli adalah *software access controller open source* berbasis pada *Chillispotproject* yang sekarang sudah tidak aktif. Lebih tepatnya, *Coovachilli* aktif dikembangkan oleh kontributor *Chillispot*. *CoovaChilli* juga bisa diartikan sebagai sebuah *software access control* yang kaya akan fitur, yang dapat memberikan *captive portal* dan menggunakan RADIUS untuk mengontrol akses dan akunting. *CoovaChilli* adalah bagian integral dari *firmware CoovaAP OpenWRT*.

Coovachilli akan mengambil alih kontrol dari internal *interface* (eth1) menggunakan *socketraw promiscuous*. *Coovachilli* kemudian akan menggunakan kernel modul *vtun* untuk membangun *interface virtual* untuk meneruskan paket yang diterima atau dikirim ke WAN. Pada dasarnya, kernel modul *vtun* digunakan untuk memindahkan paket IP dari kernel ke *mode user*, hingga *Coovachilli* dapat berfungsi tanpa *non-standard kernel module*. *Coovachilli* kemudian akan memberikan DHCP, ARP dan HTTP *Hijacking* pada *interface* “dhcpif”, contohnya pada eth1.

Client akan tersambung ke *interface* ini akan dibatasi oleh *captive portal* sampai di otorisasi. *Client* hanya akan dapat DNS dan *web site* yang sudah di *approved* oleh “walled garden”. Autentikasi dan otorisasi di *Coovachilli* akan dilakukan menggunakan salah satu dari dua cara berikut. Apakah itu menggunakan *MAC based authentication* (menggunakan pilihan *macauth* di *chilli.conf*). Metode yang kedua menggunakan *captive portal* yang akan menginisiasi proses autentikasi. Saat sebuah *client* yang tidak terautentikasi berusaha untuk mengakses *web* (pada *port* 80), permohonan untuk menyambung ke *web* akan ditangkap oleh *Coovachilli* dan akan di *direct* ke *captive portal*.

Hotspotlogin.cgi akan menyajikan halaman ke *user* dengan kolom *username* dan *password*. Data autentikasi ini akan di *forward* ke *FreeRADIUSserver*, yang akan mencocokkan informasinya ke *back end* menggunakan PAP, CHAP atau MSCHAPv2. *FreeRADIUSback end* disini adalah *MySQL*.

User kemudian akan ditentukan apakah ditolak atau diijinkan oleh *FreeRADIUS*, yang akan disampaikan oleh *hotspotlogin.cgi* dalam bentuk *message*/pesan penolakan atau pemberitahuan sudah “sukses”. [5]

2.9 RADIUS

RADIUS adalah sebuah protokol keamanan komputer yang berguna untuk melakukan autentikasi, otorisasi dan akuntansi pengguna secara terpusat guna mengakses jaringan. RADIUS didefinisikan di dalam *RFC 2865* dan *RFC 2866*, yang pada awalnya digunakan untuk melakukan autentikasi terhadap akses jaringan secara jarak jauh dengan menggunakan *dial-up*. RADIUS kemudian diimplementasikan untuk melakukan autentikasi terhadap akses jaringan secara jarak jauh dengan menggunakan koneksi selain *dial-up*, seperti halnya *Virtual Private Networking (VPN)*, *access point* nirkabel, *switch Ethernet*, dan perangkat lainnya.

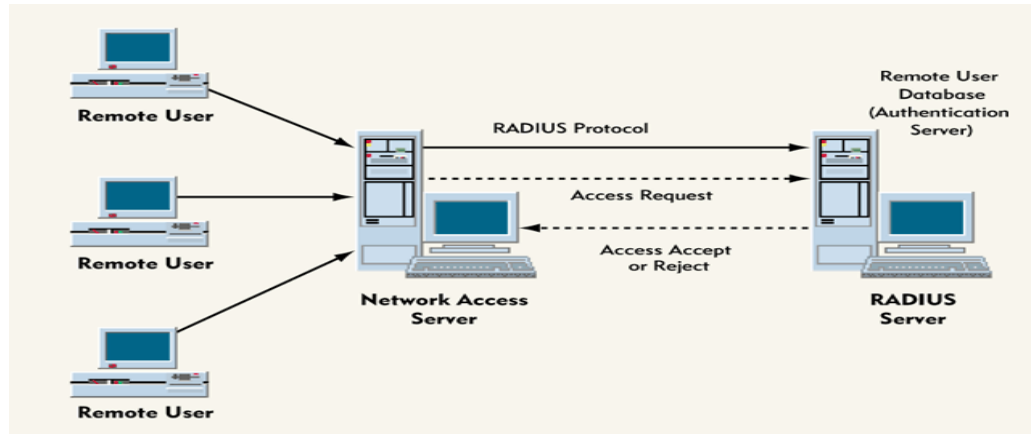
RADIUS banyak dipakai oleh *provider* dan *Internet Service Provider (ISP)* untuk autentikasi dan *billing*-nya. RADIUS juga bisa dipakai oleh jaringan Kejaksaan untuk autentikasi para penggunanya dan untuk mengamankan jaringan Kejaksaan yang ada. Di Indonesia sudah ada *service* RADIUS, namun berbayar seperti *indohotspot.net*. ada juga *service* yang tidak berbayar, dan dikelola oleh luar negeri seperti *chilidog.org*.

RADIUS *Server* memiliki banyak keunggulan, selain kontrol menjadi terpusat, anda juga bias merangkum dan membuat beberapa *report access* atau trafik dengan mudah. Selain itu banyak sekali yang menjual dan menyediakan pengontrol RADIUS *Server*, sehingga bisa di kontrol dari *web* dan dimanapun anda berada.

RADIUS *Server* sangat berguna untuk mengontrol *user-user* yang jauh dari jangkauan, jadi kita hanya pantau dari *Server* RADIUS tersebut untuk *security authentic*-nya. Selain kita bisa mengontrol *user* kita bisa kontrol semua *limit traffic* yang akan dilewatkan ke *end user*. [6]

2.9.1 Cara kerja RADIUS

RADIUS menggunakan konsep AAA (*Authentication, Authorization, Accounting*). Konsep tersebut dapat dilihat pada gambar 2.10.



Gambar 2.10 Aliran Pada RADIUS

Penjelasan dari gambar 2.10 aliran pada RADIUS:

1. *Access Request*
Pesan ini dibuat oleh *client* RADIUS yang dikirimkan ke server untuk meneruskan permintaan dari *user* yang meminta autentikasi dan otorisasi untuk percobaan koneksi.
2. *Access Accept*
User akan diberikan akses masuk ketika *user* terautentikasi, server RADIUS akan sering mengecek agar *user* hanya menggunakan sumber daya sesuai yang diminta. Informasi tentang *user* disimpan dalam *database* melalui LDAP (*Leightweight Directory Access Protocol*).
3. *Access Reject*
User ditolak aksesnya ke semua sumber daya dalam jaringan. Penyebabnya bisa termasuk kegagalan untuk menyediakan identifikasi yang tepat atau nama akun yang salah/tidak terdaftar.

Authentication adalah suatu proses dimana *user* diidentifikasi oleh server AAA sebelum *user* menggunakan jaringan. Pada proses ini, *user* meminta hak akses kepada *Network Access Server* (NAS) untuk menggunakan suatu jaringan. NAS kemudian menanyakan kepada server AAA apakah *user* yang bersangkutan berhak untuk menggunakan jaringan atau tidak.

Authorization adalah pengalokasian layanan apa saja yang berhak diakses oleh *user* pada jaringan. *Authorization* dilakukan ketika *user* telah dinyatakan berhak untuk menggunakan jaringan.

Accounting merupakan proses yang dilakukan oleh NAS dan AAA server yang mencatat semua aktivitas *user* dalam jaringan, seperti kapan *user* mulai menggunakan jaringan, kapan *user* mengakhiri koneksinya dengan jaringan, berapa lama *user* menggunakan jaringan, berapa banyak data yang diakses *user* dari jaringan, dan lain sebagainya. Informasi yang diperoleh dari proses *accounting* disimpan pada AAA server, dan dapat digunakan untuk berbagai keperluan seperti *billing*, *auditing*, atau manajemen jaringan.[7]

2.9.2 FreeRADIUS

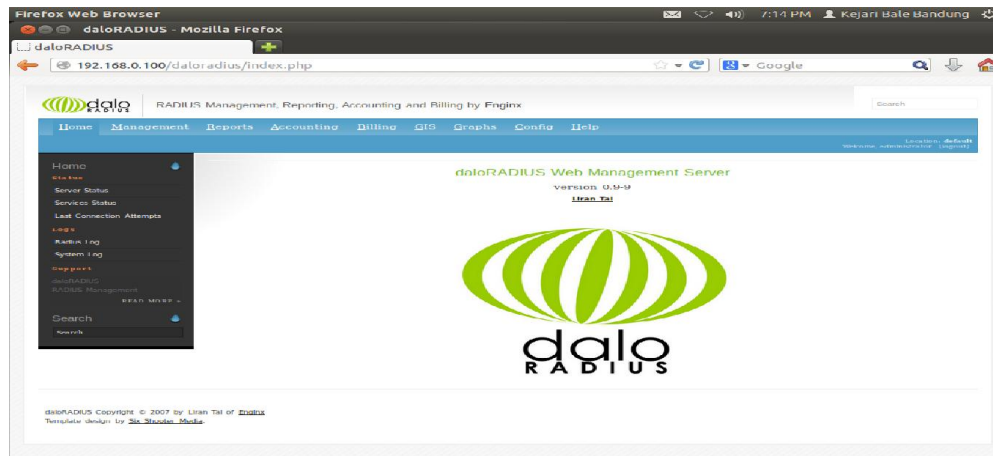
FreeRADIUS adalah sebuah produk RADIUS *open-source* yang dilengkapi dengan modul, performa tinggi dan banyak fitur lain termasuk didalamnya adalah *server*, *client*, *libraries* dan berbagai kegunaan tambahan yang berhubungan dengan RADIUS.

Sebagai *open-source* RADIUS yang pertama, *FreeRADIUS* ini dimasukkan sebagai paket standar dengan banyak sistem operasi, memiliki paket *binary*, dan sumber yang bisa digunakan untuk membangun hampir disemuanya. Penyebaran produksi ini mencakup instalasi berskala besar yang berisikan banyak *server* AAA dengan pengguna lebih dari sepuluh juta jutaan permintaan setiap harinya. *FreeRADIUS* juga mendukung permintaan *proxy* dengan *fail-over* (kemampuan untuk berpindah ke *server* yang redundansi atau

siaga ketika terjadi kegagalan) dan *load balancing* (teknik mendistribusikan pekerjaan pada *port-port* jaringan untuk meningkatkan penggunaan jaringan).[8]

2.9.3 *DaloRADIUS*

DaloRADIUS merupakan *platformweb* untuk mengontrol *FreeRADIUS* yang bertujuan mengelola *user*, pelaporan grafis dan akunting. *DaloRADIUS* ditulis dengan bahasa pemrograman PHP, javascript dan menggunakan sistem abstraksi *database* yang berarti bahwa *DaloRADIUS* mendukung banyak sistem basis data diantaranya *MySQL* yang paling populer, *PostgreSQL*, *SQLite*, *MSSQL* dan banyak lainnya. Hal ini didasarkan pada penyebaran *freeradius*, dengan *database server* yang berfungsi sebagai *backend*. [9]



Gambar 2.11 Tampilan *DaloRADIUS*

2.10 *Web Server Apache*

Web Server adalah sebuah perangkat lunak server yang berfungsi menerima permintaan HTTP atau HTTPS dari *client* yang dikenal dengan *web browser* dan mengirimkan kembali hasilnya dalam bentuk halaman-halaman *web* yang umumnya berbentuk dokumen HTML.

Apache adalah *serverweb* yang dapat dijalankan dibanyak sistem operasi (*Unix, BSD, Linux, Microsoft Windows, Novell Netware* dan platform lainnya) yang berguna untuk melayani dan mengfungsikan situs *web*.

Protokol yang digunakan untuk melayani fasilitas *web* atau *www* (*world wide web*) ini menggunakan HTTP. *Apache* memiliki fitur-fitur canggih seperti pesan kesalahan yang dapat dikonfigurasi, autentikasi berbasis data dan lainnya. *Apache* juga didukung oleh sejumlah antarmuka pengguna atau tampilan berbasis grafik (GUI) yang memungkinkan penanganan server menjadi mudah.[10]

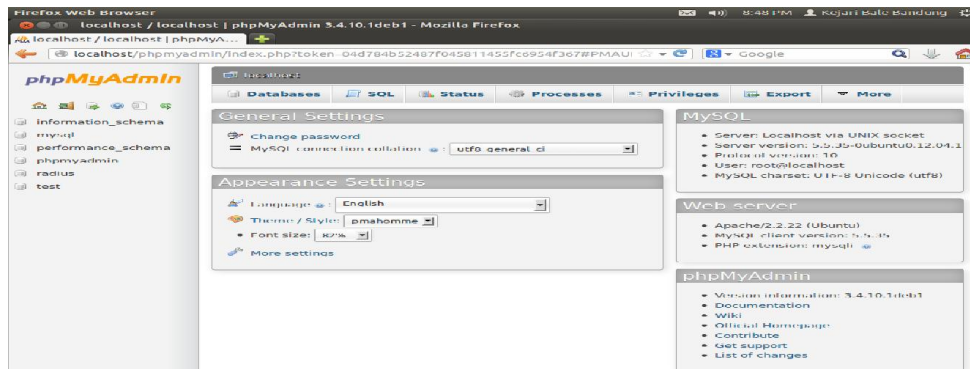
2.11 MySQL

MySQL adalah *Relational Database Management System* (RDBMS) yang didistribusikan secara gratis dibawah lisensi GPL (*General Public License*). Dimana setiap orang bebas untuk menggunakan *MySQL*, namun tidak boleh dijadikan produk turunan yang bersifat komersial. *MySQL* sebenarnya merupakan turunan salah satu konsep utama dalam *database* sejak lama, yaitu *SQL* (*Structured Query Language*). *SQL* adalah sebuah konsep pengoprasian *database*, terutama untuk pemilihan atau seleksi dan pemasukkan data, yang memungkinkan pengoprasian data dikerjakan dengan mudah secara otomatis. Keandalan suatu sistem *database* (DBMS) dapat diketahui dari car kerja *optimizer*-nya dalam melakukan proses perintah-perintah *SQL*, yang dibuat oleh *user* maupun program-program aplikasinya. Sebagai *database server*, *MySQL* dapat dikatakan lebih unggul dibandingkan *database server* lainnya dalam *quwery* data.[10]

2.12 PhpMyAdmin

PhpMyAdmin merupakan halaman yang terdapat pada *webserver*. Fungsi dari halaman ini adalah sebagai pengendali *databaseMySQL* menggunakan *webserver*. *PhpMyAdmin* mempermudah seorang pengelola *database* karena

tampilan yang *user-friendly* di akses melalui *web browser* dibanding tampilan bawaan *MySQL* yang hanya berupa perintah-perintah (*command*). [10]



Gambar 2.12 Tampilan *phpMyAdmin*