BAB II
TEORI PENUNJANG

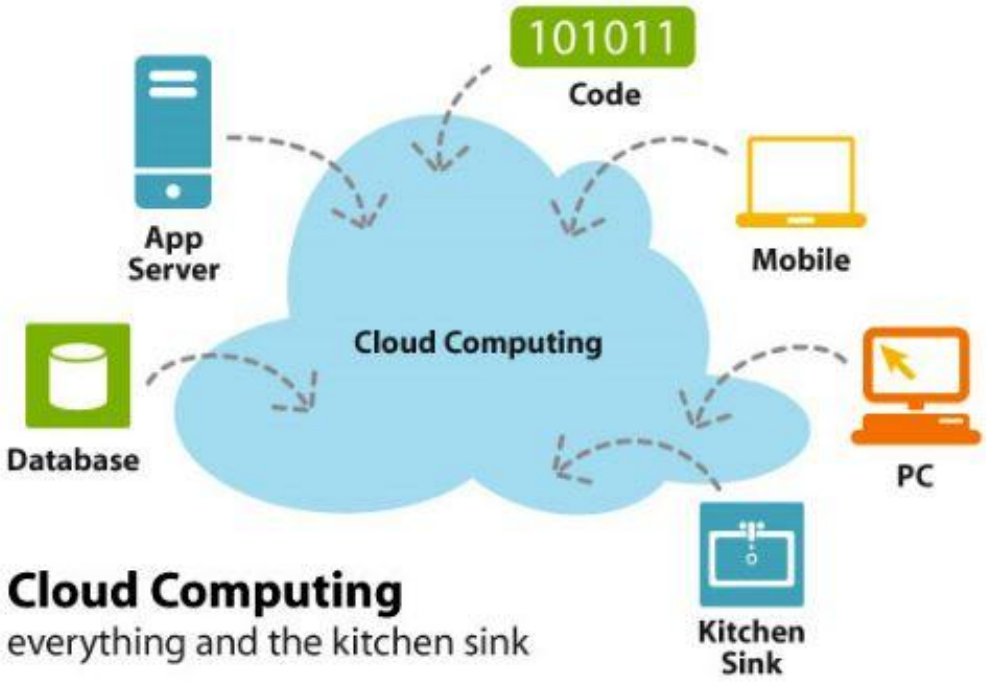
2.1 *Cloud Computing*

Cloud computing adalah gabungan pemanfaatan teknologi komputer (komputasi) dan pengembangan berbasis internet. *Cloud* adalah metafora dari internet, sebagaimana *cloud* yang sering digambarkan di dalam jaringan komputer tersebut, *cloud* dalam *cloud computing* juga merupakan abstraksi dari infrastruktur kompleks yang disembunyikannya. Ia adalah suatu metoda komputasi dimana kapabilitas terkait teknologi informasi disajikan sebagai layanan (*as a service*), sehingga pengguna dapat mengaksesnya lewat internet tanpa mengetahui apa yang ada di dalamnya, ahli dengannya, atau memiliki kendali terhadap infrastruktur teknologi yang membantunya.[1]

Cloud computing merupakan suatu paradigma dimana informasi secara permanen tersimpan di server (di internet) dan tersimpan secara sementara di komputer pengguna (*client*) termasuk di dalamnya adalah *desktop*, komputer *tablet*, *notebook*, *smartphone* dan sensor-sensor lain. Pernyataan ini merupakan definisi *cloud computing* yang dipublikasikan oleh standar IEEE.

Lembaga riset Gartner Group juga menyatakan bahwa *cloud computing* merupakan wacana yang tidak boleh dilewatkan oleh seluruh organisasi IT ataupun praktisi IT yang berkepentingan di dunia IT, mulai saat ini dan dalam beberapa waktu mendatang. Ini disebabkan karena *cloud computing* adalah sebuah mekanisme yang memungkinkan kita "menyewa" sumber daya teknologi informasi (*software, processing power, storage*, dan lainnya) melalui internet dan memanfaatkan sesuai kebutuhan kita dan membayar sesuai dengan yang digunakan oleh kita saja. Dengan konsep ini, maka semakin banyak orang yang bisa memiliki akses dan memanfaatkan sumber daya tersebut, karena tidak harus melakukan investasi besar-besaran. Apalagi dalam kondisi ekonomi seperti sekarang, setiap organisasi akan berpikir panjang untuk mengeluarkan investasi tambahan di sisi IT. Terlebih hanya untuk mendapatkan layanan-layanan yang

mungkin hanya dibutuhkan sewaktu-waktu saja. Gambar dibawah ini merupakan ilustrasi dari *cloud computing*. [2]



Gambar 2.1. Ilustrasi Dari *Cloud Computing* [2]

2.1.1 Karakteritik *Cloud Computing*

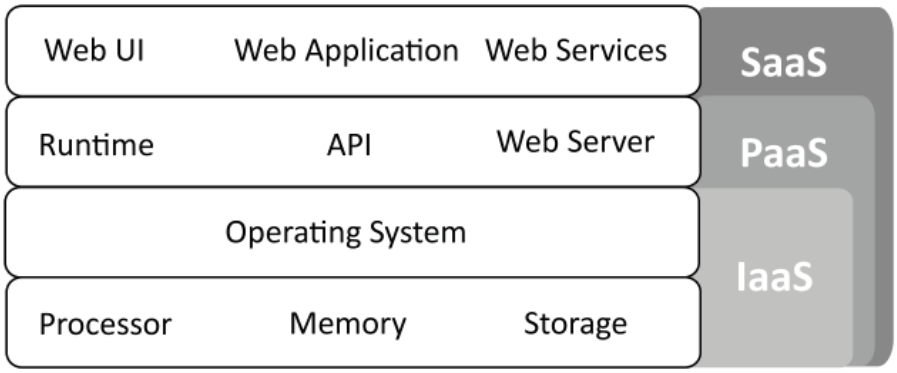
Ada beberapa karakteristik yang harus dipenuhi agar layanan yang ada di internet dikatakan sebagai layanan *cloud computing* diantaranya bersifat:

1. *On-Demand* : Pengguna dapat berlangganan hanya yang dibutuhkan saja dan membayar sesuai apa yang digunakan saja.
2. Elastis : Pengguna dapat menambah atau mengurangi jenis dan kapasitas layanan yang dinginkan kapan saja dan sistem selalu bisa mengakomodasi perubahan tersebut.
3. *Multi-Tenant* : Memungkinkan sejumlah sumber daya komputasi digunakan secara bersama-sama oleh sejumlah *user*, dimana sumber daya tersebut baik yang berbetuk fisik atau virtual, dapat dialokasikan secara dinamis untuk kebutuhan pengguna sesuai permintaan.

4. Layanan sepenuhnya dikelola oleh *provider*, yang dibutuhkan oleh pengguna hanyalah komputer personal/*notebook* ditambah koneksi internet.
5. Akses Pita Lebar : Layanan yang terhubung melalui jaringan pita lebar, terutama dapat diakses secara memadai memalui jaringan internet. Baik menggunakan *thin client*, *thick client*, ataupun media lain seperti *smartphone*.
6. Layanan yang terukur (*Measured Service*) : Sumber daya *cloud* yang tersedia harus dapat diatur dan dioptimasi penggunaannya, dengan suatu sistem pengukuran yang dapat mengukur penggunaan dari setiap sumber daya komputasi yang digunakan (penyimpanan, *memory*, *processor*, lebar pita, aktivitas *user*, dan lainnya). Dengan demikian, jumlah sumber daya yang digunakan dapat secara transparan diukur yang akan menjadi dasar bagi *user* untuk membayar biaya penggunaan layanan.[2]

2.1.2 Jenis-Jenis Layanan *Cloud Computing*

Layanan cloud komputing dapat dikategorikan kedalam 3 lapisan, dapat dilihat pada gambar dibawah ini.



Gambar 2.2. Lapisan *Cloud Computing* [3]

Penjelasan mengenai lapisan *cloud computing* dikemukakan berdasarkan dari Gambar 2.2. yaitu:

1. *Software as a Service* : Merupakan layanan dimana pengguna dapat mengekseskusi suatu program aplikasi tanpa perlu menginstal aplikasi tersebut, namun cukup dapat mengaksesnya melalui internet.
2. *Platform as a Service* : Memungkinkan *developer* untuk membangun suatu aplikasi diatas suatu platform yang dapat dikostumisasi sehingga spesifikasi perangkat client bukanlah lagi masalah untuk membangun dan menjalankan aplikasi dengan performansi apapun.
3. *Infrastructure as a Service* : IaaS membungkus perangkat keras (*processor, memory, storage*) dan sistem operasi yang merupakan antarmuka antara CPU dengan perangkat menjadi sebuah paket layanan. Layanan ini dapat digunakan oleh konsumen, cukup dengan meng-*invoke*-nya melalui jaringan sehingga seolah-olah konsumen menggunakan sebuah server yang secara fisik bukan miliknya dan tidak ada di tempatnya, melainkan server yang digunakan bersama dan dijalankan secara *remote* dari tempat yang jauh.[2]

2.1.3 Ruang Lingkup *Cloud Computing*

Cloud computing ditunjukan untuk kebutuhan pengguna, maka *cloud computing* dibagi kedalam empat jenis ruang lingkup, diantaranya:

1. *Private Cloud*

   Di mana sebuah infrastruktur layanan *cloud,* dioperasikan hanya untuk sebuah organisasi tertentu. Infrastruktur cloud itu bisa saja dikelola oleh si organisasi itu sendiri atau oleh pihak ketiga. *Private cloud* sendiri sebenarnya tak ubah seperti *data center* lainnya, hanya saja *private cloud* ini satu tingkat lebih tinggi dari *data center* biasa atau bisa disebut sebagai *dynamic data center*. Bisa disebut demikian karena perusahaan yang memiliki *data center* sendiri dan kemudian mengadopsi teknologi *cloud* ini, dalam pengelolaannya akan dimudahkan, karena semua bisa dibuat terotomatisasi.
2. *Community Cloud*

Dalam model ini, sebuah infrastruktur *cloud* digunakan bersama-sama oleh beberapa organisasi yang memiliki kesamaan kepentingan, misalnya dari sisi misinya, atau tingkat keamanan yang dibutuhkan, dan lainnya.

3. *Public Cloud*

   Jenis *cloud* ini diperuntukkan untuk umum oleh penyedia layanannya.

4. *Hybrid Cloud*

   Untuk jenis ini, infrastruktur cloud yang tersedia merupakan komposisi dari dua atau lebih infrastruktur *cloud* (*private*, *community*, atau *public*). Di mana meskipun secara entitas mereka tetap berdiri sendiri-sendiri, tapi dihubungkan oleh suatu teknologi/mekanisme yang memungkinkan portabilitas data dan aplikasi antar *cloud* itu. Misalnya, mekanisme *load balancing* yang antar *cloud*, sehingga alokasi sumberdaya bisa dipertahankan pada level yang optimal. [4]

2.2 *Remote Authentication Dial In User Service* (RADIUS)

Remote Authentication Dial-In User Service, yang sering disingkat menjadi RADIUS, adalah sebuah protokol keamanan komputer yang digunakan untuk melakukan otentikasi, otorisasi, dan pencatatan akun pengguna secara terpusat untuk mengakses jaringan. RADIUS didefinisikan didalam RFC 2865 dan RFC 2866, yang pada awalnya digunakan untuk melakukan otentikasi terhadap akses jaringan jarak jauh (*remote*) dengan menggunakan koneksi *dial-up*.

RADIUS kini telah diimplementasikan untuk melakukan otentikasi 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 mula-mula dikembangkan oleh perusahan Livingston. Pada awal pengembangannya, RADIUS menggunakan port 1645, yang ternyata bentrok dengan layanan “datametrics”. Sekarang, port yang dipakai RADIUS adalah port 1812. Berikut ini adalah RFC (*Request For Comment*) yang berhubungan dengan RADIUS:

1. RFC 2865 : *Remote Authentication Dial-In User Service* (RADIUS)
2. RFC 2866 : RADIUS *Accounting*

3. RFC 2867 : RADIUS *Accounting for Tunneling*
4. RFC 2868 : RADIUS *Authentication for Tunneling*
5. RFC 2869 : RADIUS *Extensions*
6. RFC 3162 : RADIUS *over* IP6
7. RFC 2548 : Microsoft Vendor-Specific RADIUS *Attributes*

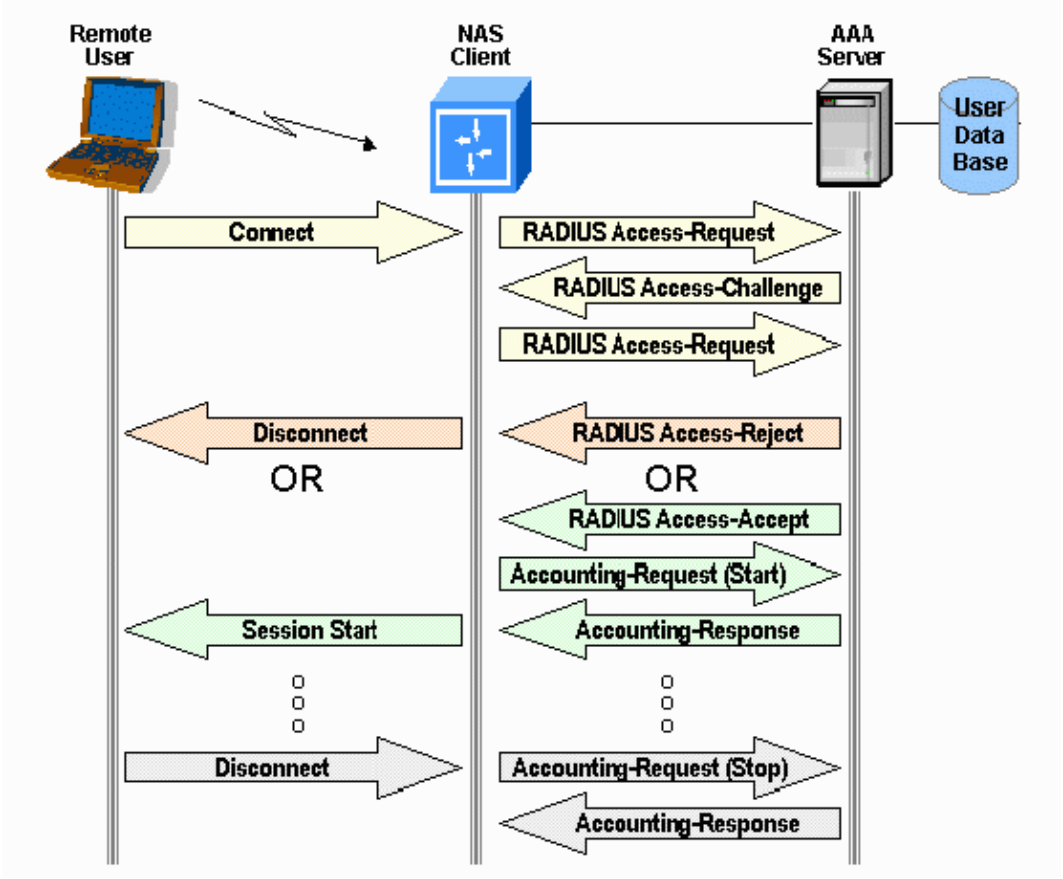
2.2.1 Karakteristik Server RADIUS

Ada beberapa karakteristik pada server RADIUS diantaranya sebagai berikut:

1. Berbasis UDP Protocol
2. Bisa ditempatkan dimana saja di internet dan dapat membuat autentikasi (PPP, PAP, CHAP, MS-CHAP, EAP) antara *Network Access Server* (NAS) dan server itu sendiri.
3. RADIUS menggunakan Remote Access Server (RAS) Secure ID untuk membuat autentikasi yang kuat dalam pengontrolan akses

2.2.2 Cara Kerja RADIUS

RADIUS menggunakan konsep AAA (*Authentication*, *Authorization*, *Accounting*). Konsep tersebut dapat digambarkan sebagai berikut:



Gambar 2.3. Aliran Data Pada RADIUS

1. *Access-Request*

   Pesan ini dibuat oleh client RADIUS yang dikirimkan ke server untuk meneruskan permintaan dari *user* yang meminta otentikasi dan otorisasi untuk percobaan koneksi.

2. *Access-Accept*

   User diberikan akses masuk ketika *user* terotentikasi, server RADIUS akan sering mengecek agar *user* hanya menggunakan sumber daya sesuai dengan yang diminta. Misalnya *user* hanya boleh mengakses fasilitas *hotspot*, dan tidak untuk menggunakan *printer*. Informasi tentang *user* dalam *database* bisa disimpan secara lokal dalam server RADIUS atau disimpan dalam tempat penyimpanan eksternal seperti LDAP atau *Active Directory*

3. *Access-Reject*

User ditolak aksesnya ke semua sumber daya dalam jaringan. Penyebabnya bisa termasuk kegagalan untuk menyediakan indentifikasi yang tepat atau nama akun yang salah/tidak aktif

4. *Access-Challenge*

   Permintaan informasi tambahan dari *user*, seperti *password* alternatif, PIN, token atau kartu.

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.[5]

2.3 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 berbasiskan 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, dsb sehingga tidak perlu memasang program *antivirus*.
3. Sangat ringan dan boleh digunakan pada komputer dengan *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*, mendengar musik, memainkan *video, photo viewer*, kalkulator dsb
6. Terdapat Live CD, yang memperbolehkan pengguna untuk mencoba menggunakan Ubuntu tanpa perlu installasi ke dalam *harddisk*, hanya perlu boot menggunakan CD Ubuntu untuk masuk ke *Live CD session.*
7. Ada Ubuntu *Software Center*, yang memudahkan pengguna untuk mengunduh aplikasi lainnya yang tidak ada pada Ubuntu.
8. Cocok untuk 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 sang pengguna harus menambahkan *plugin* untuk mp3 tersebut. Namun jika ingin di instal secara *offline*, pengguna harus

mempunyai CD/DVD Repository yang berisi segala kebutuhan yang terdapat di dalamnya.

2. Koneksi Internet: Dalam penginstalan *software-software* tambahan dan pluginnya, ubuntu sangat memerlukan koneksi internet. Ubuntu yang telah kita install terhubung dengan *mirror* untuk installasi tambahan, *update* dan sebagainya. Pada saat kita akan menambahkan aplikasi kedalamnya melalui *Synaptic Package Manager*, atau install manual melalui Terminal, nanti akan melewati *mirror site* yang terdaftar di OS tersebut, kemudian di unduh aplikasi tersebut.
3. Tidak *User Friendly*: Mungkin inilah salah satu kekurangan ubuntu bagi para penggunanya. Dari segi penempatan menu, *management file* & *properties*nya, aplikasi – aplikasi di dalamnya dan lainnya, akan membuat kaku para penggunanya karena ketidakbiasaan dalam menjalankannya. Dalam menggunakannya 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*).[6]

Berikut adalah tampilan dari ubuntu versi 11.10



Gambar 2.4. Tampilan Ubuntu 11.10

2.4 *FreeRADIUS*

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

Sebagai *open-source* RADIUS yang pertama, FreeRADIUS ini dimasukan sebagai paket standar dengan banyak sistem operasi, memiliki paket *binary*, dan sumber yang bisa digunakan untuk membangun hampir disemuanya. Penyebaran produksi ini mencangkup instalasi berskala besar yang berisikan banyak server AAA dengan pengguna lebih dari sepuluh juga jutaan permintaan setiap harinya. FreeRADIUS juga mendukung permintaan *proxy* dengan *fail-over* (kemampuan untuk berpindah ke server yang redudansi atau siaga ketika terjadi kegagalan) dan *load balancing* (teknik mendistribusikan pekerjaan pada port-port jaringan untuk meningkatkan penggunaan jaringan). Sebaik dengan kemampuan untuk mengakses berbagai macam tipe basis data *back-end*. Perbedaan yang dimiliki antara FreeRADIUS dengan server RADIUS yang lain:

- Akses didasarkan dengan *huntgroup*.
- Banyak *entry* DEFAULT di *file* raddb/users.
- Setiap *entry* file pengguna memiliki pilihan "*fall through*".
- Menyimpan/meng-*cache*-kan semua file *config* di memori.
- Menyimpan daftar dari pengguna yang masuk (*file* radutmp).
- Program "radwho" dapat diinstal sebagai "fingerd".
- Merekam *format file* UNIX wtmp dan detail rekaman data RADIUS.
- Mendukung *Simultaneous-Use*=X parameter. Yang berarti bahwa *login* berkali-kali dapat dihindari.
- Mendukung *atribut* yang terspesifikasi dari vendor, termasuk USR *login* berkali-kali dapat dihindari.
- Mendukung atribut yang yang terspesifikasi dari vendor, termasuk USR yang tidak standar.
- Mendukung *proxy.*
- Mendukung paket "*alive*"

- *Exec-Program-Wait* mengizinkan dibuatnya sebuah program eksternal yang dijalankan sesudah otentikasi dan menghasilkan sebuah daftar dari pasangan A/V yang kemudian ditambah ke respon.
- Mendukung *Pluggable Authentication Modules* (PAM).[7]

2.5 *Web Server Apache*

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

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

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

Apache merupakan perangkat lunak sumber terbuka (*open source*) dikembangkan oleh komunitas terbuka yang terdiri dari pengembangan-pengembangan dibawah naungan *Apache Software Foundation.*[8]

2.6 MySQL

MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL (bahasa Inggris: *database management system*) atau DBMS yang *multithread*, *multi-user*, dengan sekitar 6 juta instalasi di seluruh dunia. MySQL AB membuat MySQL tersedia sebagai perangkat lunak gratis dibawah lisensi GNU *General Public License* (GPL), tetapi mereka juga menjual dibawah lisensi komersial untuk kasus-kasus dimana penggunaannya tidak cocok dengan penggunaan GPL.

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 pengoperasian *database*, terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis. Keandalan suatu sistem *database* (DBMS) dapat diketahui dari cara 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 query data. Hal ini terbukti untuk *query* yang dilakukan oleh *single user*, kecepatan *query* MySQL bisa sepuluh kali lebih cepat dari PostgreSQL dan lima kali lebih cepat dibandingkan Interbase.

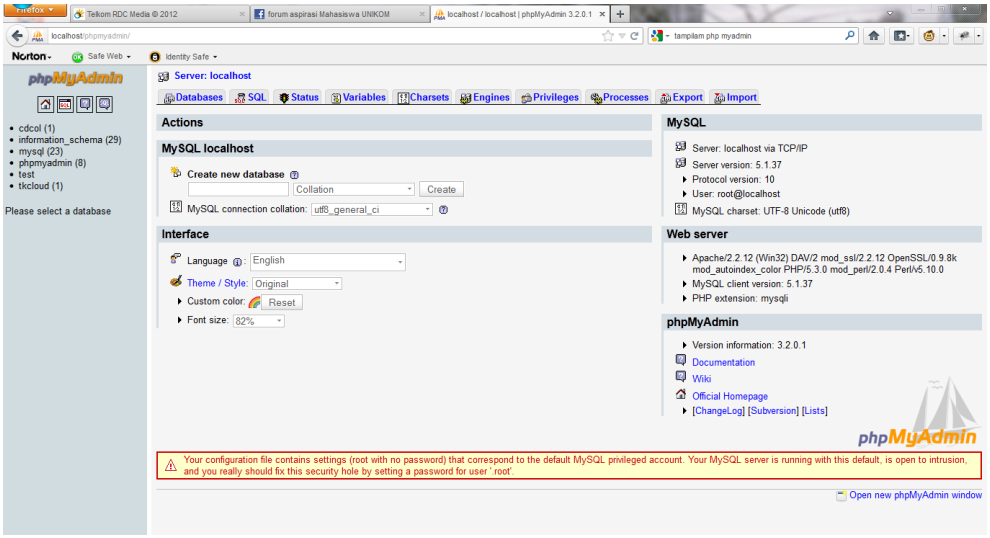
MySQL memiliki beberapa keistimewaan, antara lain :

1. Portabilitas. MySQL dapat berjalan stabil pada berbagai sistem operasi seperti Windows, Linux, FreeBSD, Mac Os X Server, Solaris, Amiga, dan masih banyak lagi.
2. *Open Source*. MySQL didistribusikan secara open source, dibawah lisensi GPL sehingga dapat digunakan secara cuma-cuma.
3. *Multi-User*. MySQL dapat digunakan oleh beberapa *user* dalam waktu yang bersamaan tanpa mengalami masalah atau konflik.
4. *Performance Tuning*. MySQL memiliki kecepatan yang menakjubkan dalam menangani *query* sederhana, dengan kata lain dapat memproses lebih banyak SQL per satuan waktu.
5. Jenis Kolom. MySQL memiliki tipe kolom yang sangat kompleks, seperti *signed*/*unsigned integer*, *float*, *double*, *char*, *text*, *date*, *timestamp*, dan lainnya.
6. Perintah dan Fungsi. MySQL memiliki operator dan fungsi secara penuh yang mendukung perintah *Select* dan *Where* dalam perintah (*query*).
7. Keamanan. MySQL memiliki beberapa lapisan sekuritas seperti level *subnetmask*, nama host, dan izin akses *user* dengan sistem perizinan yang mendetail serta sandi terenkripsi.

8. Skalabilitas dan Pembatasan. MySQL mampu menangani basis data dalam skala besar, dengan jumlah rekaman (*records*) lebih dari 50 juta dan 60 ribu tabel serta 5 milyar baris. Selain itu batas indeks yang dapat ditampung mencapai 32 indeks pada tiap tabelnya.
9. Konektivitas. MySQL dapat melakukan koneksi dengan klien menggunakan protokol TCP/IP, *Unix soket* (UNIX), atau *Named Pipes* (NT).
10. Lokalisasi. MySQL dapat mendeteksi pesan kesalahan pada klien dengan menggunakan lebih dari dua puluh bahasa. Meski pun demikian, bahasa Indonesia belum termasuk di dalamnya.

2.7 *PhpMyAdmin*

PhpMyAdmin merupakan halaman yang terdapat pada *web server*. Fungsi dari halaman ini adalah sebagai pengendali *database* MySQL menggunakan *web server*. *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*).[8]



Gambar 2.5. Tampilan *PhpMyAdmin* di Windows

2.8 *CoovaChilli*

CoovaChilli ini adalah *software access controller open source*, berbasis pada *Chillispot project* 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* / *walled-garden environment* 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 *socket raw promiscuous*. *CoovaChilli* kemudian akan menggunakan kernel modul vtun untuk membangun *interface* virtual untuk meneruskan paket yang di terima atau di kirim ke WAN. Pada dasarnya, kernel modul vtun digunakan untuk memindahkan paket IP dari kernel ke *mode user*, sedemikkan hingga *CoovaChilli* dapat berfungsi tanpa *non-standard kernel module*. *CoovaChilli* kemudian akan memberikan DHCP, ARP, and HTTP *Hijacking* pada *interface* "dhcpif", contohnya pada eth1.

Sebuah *client*/laptop akan tersambung ke *Interface* ini akan di batasi oleh "*walled garden*" sampai di otorisasi. *Client*/laptop hanya akan dapat memperoleh DNS dan Web site yang sudah di *approved* oleh "*walled garden*". Otentikasi 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) atau menggunakan cara "*Universal Access Method*" (UAM). Metoda yang ke dua menggunakan *captive portal* yang akan menginisiasi proses otentikasi. Saat sebuah *client* yang tidak terotentikasi berusaha untuk mengakses web (pada port 80), permohonan untuk menyambung ke web akan di tangkap oleh *CoovaChilli* dan akan di *redirect* ke *captive portal*.

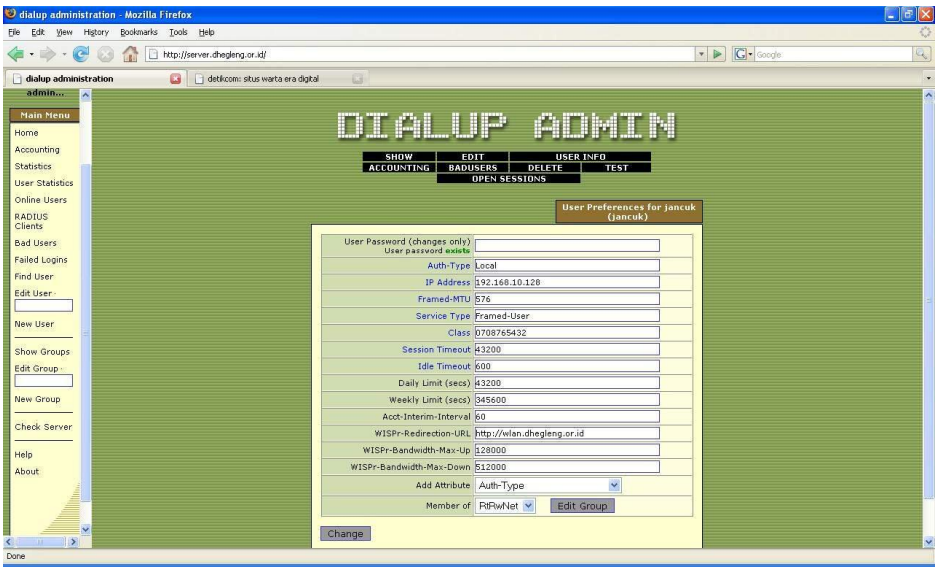
Hotspotlogin.cgi akan menyajikan halaman ke *user* dengan kolom *username* dan *password*. Data otentikasi ini akan di *forward* ke *FreeRADIUS server*, yang akan kemudian mencocokkan informasinya ke *back end* menggunakan PAP, CHAP, atau MSCHAPv2. *FreeRADIUS back-end* disini adalah MySQL, tentunya bisa servis lain seperti LDAP, *Kerberos*, *unix passwd file* atau *Active Directory*.

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'.[10]

2.9 *DialupAdmin* dan *DaloRADIUS*

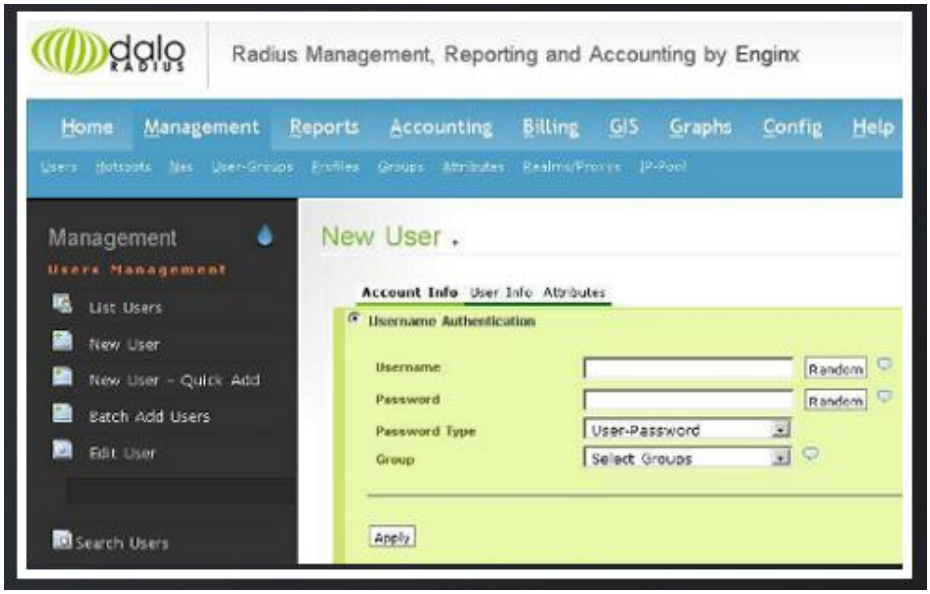
DialupAdmin adalah *tool management user* RADIUS berbasis PHP yang khusus digunakan untuk mengontrol *FreeRADIUS*. *Dialup admin* juga digunakan untuk memonitoring aktifitas klien dan melihat trafik serta data statistik penggunaan dari seorang klien *user*.

Karena dialup admin berada pada *root direktory* setelah *root folder website* yang dipakai untuk klien maka harus ada pembatasan akses khusus untuk *dialup admin* ini dengan menerapkan konfigurasi *utility htpaswwd* yang sudah disediakan oleh *Web Server Apache*. Berikut adalah tampilan dialupadmin di *web browser*.



Gambar 2.6. Tampilan *Dialup Admin*

DaloRADIUS merupakan *platform web* untuk mengontrol *FreeRADIUS* bertujuan mengelola *user*, pelaporan grafis, akunting dan dapat terintegrasi dengan *GoogleMaps* untuk Geolokasi (GIS). *DaloRADIUS* ditulis dengan bahasa pemrograman PHP, javascript dan menggunakan sistem abtraksi *database*, mendukung banyak sistem *database* diantaranya MySQL yang paling popular, PostgreSQL, SQLite, MSSQL, dan banyak lainnya.[11]



Gambar 2.7. Tampilan *User Interface DaloRADIUS* [11]