BAB II
TINJAUAN PUSTAKA

2.1 Profil Tempat Penelitian Skripsi

2.1.1 Sejarah Instansi

Tahun 2008 merupakan tahun perubahan dan kebangkitan bagi unit pengelola teknologi informasi Universitas Padjadjaran. Dengan pergantian nama dari PUSDATIN (pusat data dan teknologi informasi) menjadi UPT *Development Center of Information System and Technology for Education and Management* (DCISTEM) tanggung jawab dan kepercayaan pimpinan dalam hal pengelolaan teknologi informasi berada di unit ini. Pimpinan dalam hal ini Rektor banyak menaruh harapan besar terhadap DCISTEM sebagai pengemban visi dan misi Rektor dalam pengembangan tata kelola dan sumber daya manusia melalui efektifitas dan akuntabilitas yang mampu dihadirkan melalui pemanfaatan teknologi informasi.

Kepercayaan ini mulai dijawab oleh DCISTEM dengan 2 (dua) langkah awal yang merupakan fondasi dasar yaitu infrastruktur dan aplikasi. Pengembangan infrastruktur dan aplikasi ini puncaknya akan bermuara pada *Enterprise system* yang mengintegrasikan sistem dan konsep teknologi informasi dalam satu paket yang sinergi dan menyeluruh.

DCISTEM berdiri pada tanggal 4 Januari 2008 manggantikan UP CCIT dibawah koordinasi Pembantu Rektor V bidang perencanaan, sistem informasi dan pengawasan.

Pada Tahun 2013 seiring dengan pergantiian pimpinan Universitas dalam hal ini Wakil rektor berserta struktur keorganisasiannya UPT DCISTEM berganti nama menjadi Unit Pelayanan Teknis Teknologi Informasi DCISTEM (UPT TI DCISTEM) dibawah koordinasi wakil rektor bidang Perencanaan, Sistem Informasi dan Keuangan.

2.1.2 Logo Instansi

Berikut adalah logo dari instansi tempat penulis melakukan kerja praktek yaitu logo UPT TI DCISTEM – Unpad.



Gambar 2.1 Logo Instansi

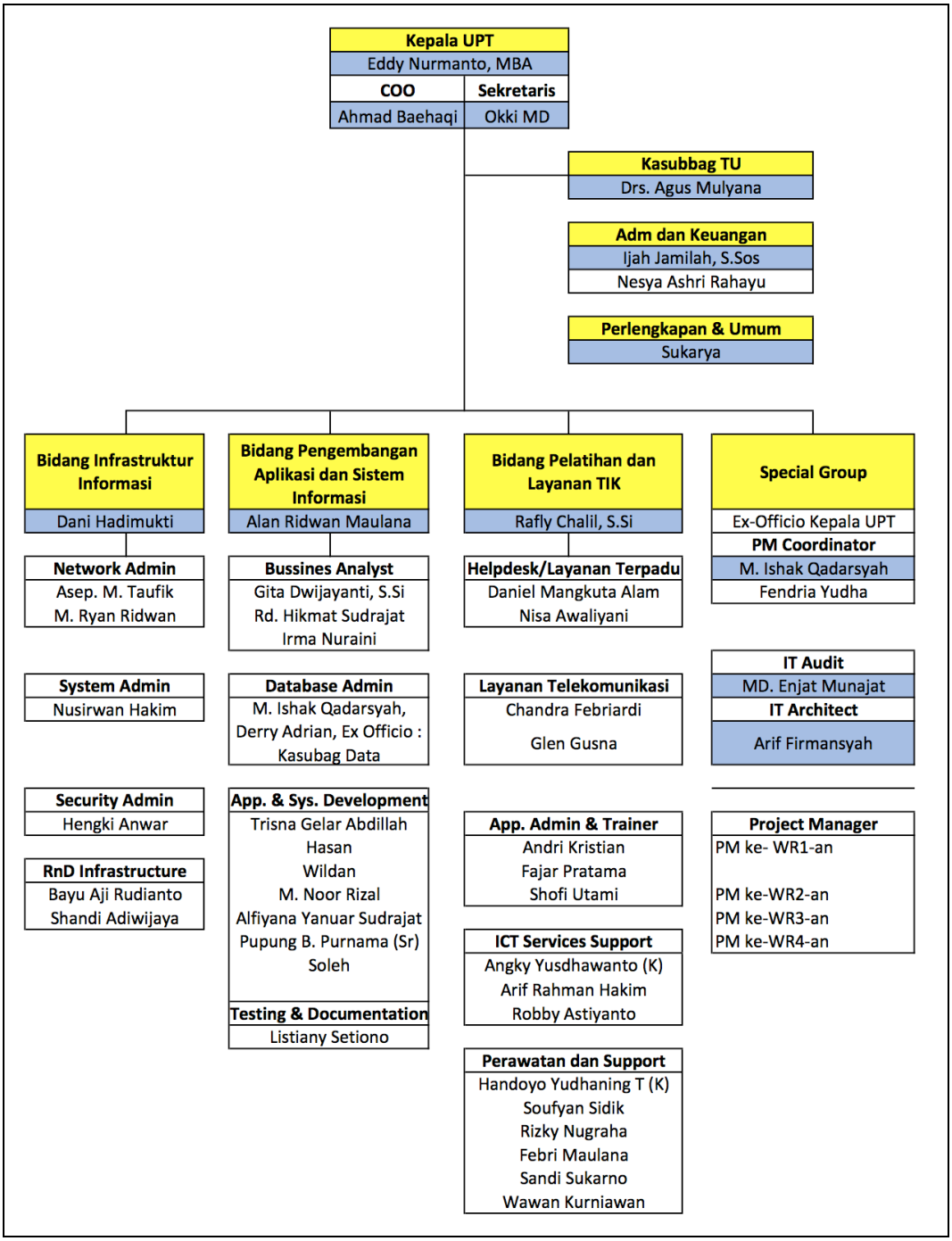
2.1.3 Badan Hukum Instansi

UPT TI DCISTEM sebetulnya tidak memiliki badan hukum secara khusus seperti perusahaan – perusahaan umumnya karena UPT TI DCISTEM ini merupakan unit pengelola dari Universitas Padjadjaran.

2.1.4 Struktur Organisasi dan *Job Description*

Struktur organisasi merupakan susunan yang terdiri dari fungsi-fungsi dan hubungan-hubungan yang ada dalam suatu instansi yang menyatakan keseluruhan kegiatan untuk mencapai sasaran instansi.

2.1.4.1 Struktur Organisasi UPT-TI



Gambar 2.2 Struktur Organisasi UPT TI

2.1.4.2 *Job Description*

Adapun *job decription* dari dari DCISTEM itu sendiri yaitu :

1. Kepala UPT DCISTEM
   a. Memimpin penyusunan perencanaan *strategic* serta rencana kegiatan dan anggaran tahunan DCISTEM untuk disahkan rektor.
   b. Memberikan arahan kepada seluruh personil baik dalam tataran strategi, istruksional, administrasi serta kendali mutu.
   c. Membangaun dan melakukan pengawasan sistem pengelolaan TIK serta penjamin mutu pelayanan TIK di lingkungan Unpad.
   d. Melakukan koordinasi pengembangan TIK dengan pimpinan serta pimpinan – pimpinan fakultas di lingkungan Unpad.
   e. Memimpin dan melaksanakan fungsi dan tugas DCISTEM.

2. *Quality Assurance*
   a. Menyusun program monitoring pengawasan kualitas layanan dan pengembangan kegiatan dan produk – produk DCISTEM.
   b. Menyusun program penjamin mutu layanan dan pengembangan kegiatan dan produk – produk DCISTEM.
   c. Melakukan pengawasan internal terhadap kualitas layanan dan pengembangan kegiatan dan produk – produk DCISTEM secara periodic.

   Melakukan penjaminan mutu internal terhadap kualitas layanan dan pengembangan kegiatan dan produk – produk DCISTEM.

3. Sekretaris *Eksekutif*
   a. Bersama dengan sekretaris II dan para koordinator bidang penyusunan perencanaan rinci berdasarkan rencana *strategic* serta rencana kegiatan dan anggaran tahunan DCISTEM untuk disahkan Rektor.
   b. Membangun hubungan dan koordinasi dengan fakultas dan lembaga di lingkungan Unpad melalui kepala unit ICT.

c. Membuat perencanaan rinci dari standar pengoperasian, pendayagunaan dan pemeliharaan serta tata kelola TIK.
d. Mengkoordinasi kegiatan masing – masing koordinator bidang serta melakukan pengawasan kegiatan wilayah kampus Unpad Jatinangor.
e. Membantu direktur dalam mengkoordinasikan dan melaksanakan fungsi dan tugas DCISTEM dalam tataran strategik dan administrasi.

4. Kepala Subbagian Tata Usaha
   a. Membina staf administrasi maupun non administrasi dalam mendukung berbagai kegiatan DCISTEM.
   b. Memproses dan mengelola administrasi dan keuangan berkaitan dengan berbagai kegiatan DCISTEM secara transparan.
   c. Bertanggungjawab mengelola seluruh urusan dokumentasi pekerjaan.
   d. Bertanggungjawab mengelola surat menyurat dan hubungan administrative dengan bagian / unit lain yang terkait.

5. Sekretaris bidang solusi perangkat lunak.
   a. Mengelola bidang solusi perangkat lunak dan kegiatan pengembangan perangkat lunak secara umum, memimpin personel divisi serta bersama – sama dengan divisi lain menentukan langkah teknis kegiatan DCISTEM.
   b. Memberikan instruksi kerja pada personel tim divisi.
   c. Menugaskan personel tim untuk melaksanakan rencana kerja sesuai dengan deskripsi kerja.
   d. Menggabungkan rencana kerja, program dan kegiatan masing – masing menjad dokumen proram kerja, untuk kemudian dimatangkan sebagai pedoman pelaksaan program kerja.

6. IT architect
   a. Menyusun rencana dan master design rencana pengembangan perangkat lunak.

b. Menganalisis kondisi, kebutuhan , dan spesifikasi sistem dalam pengembangan perangkat lunak.
c. Menyusun rencana *strategic* pengembangan perangkat lunak.
d. Melakukan pengawasan terhadap skema sistem pengembangan perangkat lunak.

7. Senior Programmer
   a. Menyusun skema teknis pengembangan perangkat lunak.
   b. Mengembangkan rencana strategi pengembangan perangkat lunak.
   c. Mendokumentasikan seluruh pekerjaan pengembangan perangkat lunak.
   d. Mengevaluasi status sistem perangkat lunak dan menyusun proses administrasi dan konfigurasi sistem perangkat lunak.

8. System Analist
   a. Menganalisis kondisi, kebutuhan, dan spesifikasi sistem dalam pengembangan perangkat lunak.
   b. Mengembangkan rencana *strategic* pengembangan perangkat lunak.
   c. Menyusun, mengembangkan, dan memodelkan *design* teknis sistem.
   d. Melakukan evaluasi terhadap skema sistem pengembangan perangkat lunak.

9. Network administrator.
   a. Memproses secara antisipatif terhadap gangguan dan keluhan pengguna TIK area Bandung dan Jatinangor.
   b. Mengembangkan dan memberikan layanan jaringan di lapangan.
   c. Melakukan pemeliharaan infrastruktur dan alat lapangan.

10. Webmaster and Designer
   a. Menyusun dan mengelola program pengembangan website dan designnya.

b. Menganalisa, menyusun, dan merancang strategi peningkatan, webometriks.
c. Bertanggung jawab mengelola dan mengawasi aksesibilitas website.
d. Bertanggung jawab menyusun program pemutakhiran data dan informasi di website.

11. Database *Analyst*
    a. Memelihara interkoneksi database agar berjalan dengan baik.
    b. Memetakan data logis ke serangkaian table dan pemeliharaan integritasnya.
    c. Melakukan koordinasi penyusunan database.
    d. Melakukan pengembangan dan adaptasi database.
    e. Mengatur kepemilikan data, hak akses.

12. Multimedia Programmer
    a. Menyusun skema teknis pengembangan aplikasi multimedia.
    b. Mengimplementasikan konfigurasi rancangan pengembangan aplikasi multimedia.
    c. Mengelola dan memelihara peralatan dan aplikasi yang digunakan dalam pengembangan aplikasi multimedia.
    d. Mengadministrasi dan mengkonfigurasi sistem perangkat lunak.
    e. Mendokumentasikan seluruh pekerjaan pengembangan aplikasi multimedia.

13. Technical Administration
    a. Melakukan pendokumentasian teknis sistem perangkat lynak.
    b. Menyediakan bantuan administrasi teknis perangkat lunak.
    c. Menyusun prosedur dan aktifitas standar teknis perangkat lunak.
    d. Berhubungan dengan pihak lain diluar Unpad terkait dengan hubungan kerja divisi.

14. Database *Administrator*
    a. Bertanggung jawab melakukan pengelolaan terhadap database.
    b. Melakukan pengisian dan pengubahan terhadap konten database.
    c. Melakukan sinkronisasi dan validasi konten database.
    d. Memelihara interkoneksi database agar berjalan dengan baik.

15. Sekretaris bidang infrastruktur informasi.
    a. Memimpin, membina, dan mengembangkan seluruh kegiatan di lingkungan koordinasi infrastruktur informasi, dan bertanggungjawab langsung kepada direktur DCISTEM Universitas Padjajaran.
    b. Koordinasi dalam pelayanan komunikasi data di lingkungan Universitas Padjajaran, meliputi perancangan, pengembangan, perbaikan, pelaporan dan pendistribusiannya.
    c. Menyusun rencana kerja dan kegiatan bidang infrastruktur informasi serta melakukan pengawasan terhadap jalannya rencana kerja dan kegiatan terkait.

       Koordinasi dengan para kepala divisi dan kepala divisi unit lain (Fakultas, Lembaga, dsb) dalam rangka korelasi dan sinkronisasi sistem informasi.

16. Network architect
    a. Menyusun kerangka dan design arsitektur jaringan.
    b. Menyusun rencana pengembangan dan pembangunan jaringan.
    c. Melakukan analisa dan menyusun rencana implementasi dan kontinuitas dari design arsitektur yang disusun.
    d. Melakukan pengawasan implementasi design arsitektur jaringan di lapangan.

17. Network administrator
    a. Bertanggung jawab mengelola sistem jaringan secara umum.
    b. Bertanggung jawab melakukan upgrade sistem *server* dan jaringan.

c. Bertanggung jawab menjaga dan mengatur arus koneksi data kearah intranet dan internet.
d. Mengembangkan dan memelihara *routing*, manajemen *bandwith* dan *firewall*.
e. Mengembangkan *active directory* dan atau aplikasi sejenisnya.

18. Network security

a. Menyusun perencanaan pengamanan sistem jaringan dan komunikasi data.
b. Bertanggung jawab dalam optimalisasi dan keamanan sistem.
c. Melakukan monitoring jalannya sistem dan memantau sisi keamanannya.
d. Melakukan inventarisasi pengamanan teknis sistem jaringan komunikasi data termasuk peralatan dan aplikasinya.

19. Network support

a. Melakukan proses instalisasi dan pengembangan fisik jaringan.
b. Melakukan pemeliharaan infrastruktur dan alat di lapanagan.
c. Mengembangkan dan memelihara *cabling, switching*, dan peralatan jaringan lainnya.
d. Mengembangkan dan memelihara konektifitas jaringan.

20. Technical support

a. Merespon secara antisipatif terhadap gangguan dan keluhan pengguna TIK area Bandung dan Jatinangor.
b. Mengembangkan dan memberikan layanan pemanfaatan jaringan di lapangan.
c. Melakukan pemeliharaan infrastruktur dan alat di lapangan.

Memberikan layanan secara langsung pada pengguna di lapangan (Fakultas dan Unit – unit kerja).

2.2 Konsep Dasar jaringan Komputer

Jaringan komputer diartikan sebagai suatu himpunan interkoneksi sejumlah komputer yang dapat saling bertukar informasi. Bentuk koneksinya tidak harus malalui kawat saja melainkan dapat menggunakan serat optik, atau bahkan satelit komunikasi.

Jaringan Komputer adalah merupakan sekelompok komputer otonom yang saling berhubungan antara satu dengan lainnya menggunakan protokol komunikasi melalui media komunikasi sehingga dapat saling berbagi informasi, program-program, dan penggunaan bersama perangkat keras seperti printer, harddisk, dan lain sebagainya. (Kristanto, 2003)

2.2.1 Jenis Jenis Jaringan Komputer

Dilihat dari ruang lingkup jangkauannya, jaringan komputer dibedakan menjadi: *LocalArea Network* (LAN), *Metropolitan Area Network* (MAN), dan *Wide Area Network* (WAN) . Sedangkan jika dilihat dari cara pengaksesan data, jaringan komputer terbagi menjadi: *Client-Server* dan *Peer to Peer.* (Wahidin, 2008)

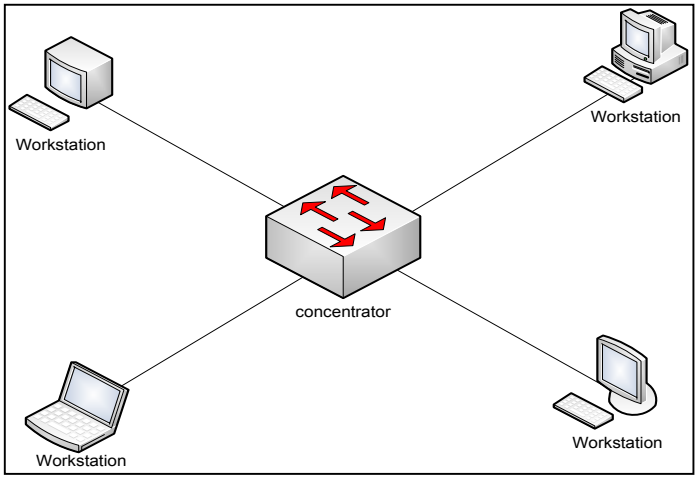
2.2.1.1 *Local Area Network* (LAN)

Local area Network (LAN), merupakan jaringan milik pribadi di dalam sebuah gedung atau kampus yang berukuran sampai beberapa kilometer. LAN seringkali digunakan untuk menghubungkan computer-komputer pribadi dan *workstation* dalam kantor perusahaan atau pabrik-pabrik untuk memakai bersama *resource* (misalnya, printer) dan saling bertukar informasi.

LAN mempunyai ukuran yang terbatas, yang berarti bahwa waktu transmisi pada keadaan terburuknya terbatas dan dapat diketahui sebelumnya. Dengan mengetahui keterbatasannya, menyebabkan adanya kemungkinan untuk menggunakan jenis desain tertentu. Hal ini juga memudahkan manajemen jaringan.

Seperti halnya saluran pelanggan telepon yang dipakai di daerah pedesaan. LAN seringkali menggunakan teknologi transmisi kabel tunggal. LAN tradisional

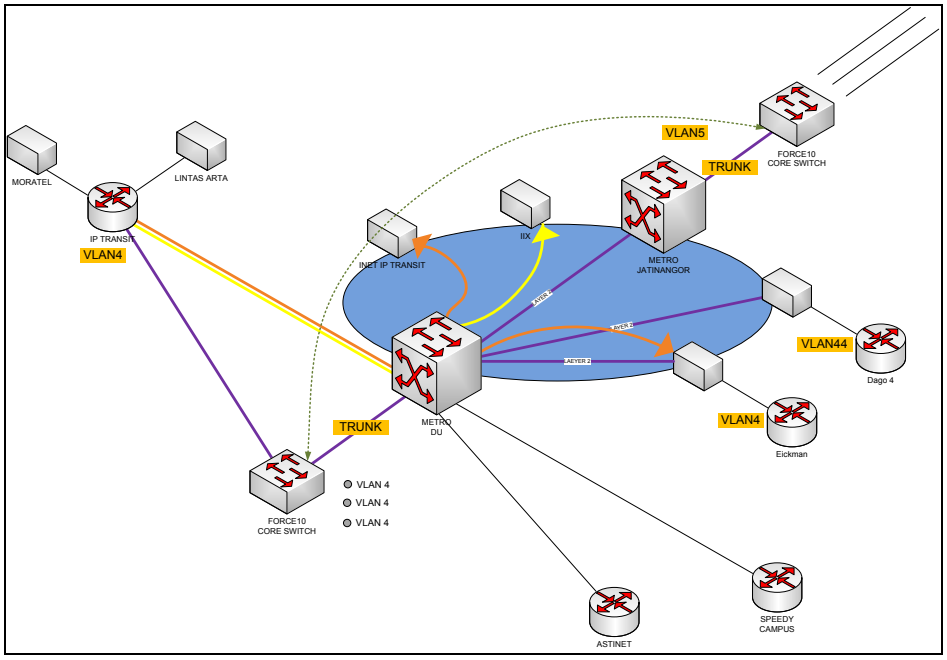
beroperasi pada kecepatan mulai 10 sampai 100 Mbps dengan *delay* rendah (puluhan *microsecond*) dan mempunyai faktor kesalahan yang kecil. LAN-LAN modern dapat beroperasi pada kecepatan yang lebih tinggi, sampai ratusan megabit/detik (Tanenbaum, 1997). Skema jaringan LAN dapat dilihat pada Gambar 2.3.



Gambar 2.3 Skema Jaringan LAN (Komputer, Konsep Jaringan Komputer dan Pengembangannya, 2003)

2.2.1.2 *Metropolitan Area Network* (MAN)

Metropolitan Area Network (MAN) pada dasarnya merupakan versi LAN yang berukuran lebih besar dan biasanya memakai teknologi yang sama dengan LAN. MAN dapat mencakup kantor-kantor perusahaan yang letaknya berdekatan atau juga sebuah kota dan dapat dimanfaatkan untuk keperluan pribadi (swasta) atau umum. MAN mampu menunjang data dan suara, dan bahkan dapat berhubungan dengan jaringan televisi kabel. MAN hanya memiliki sebuah atau dua buah kabel dan tidak mempunyai elemen *switching*, yang berfungsi untuk mengatur paket melalui beberapa kabel *output*. Adanya elemen *switching* membuat rancangan menjadi sederhana (Tanenbaum, 1997). Skema jaringan MAN dapat dilihat pada Gambar 2.4.

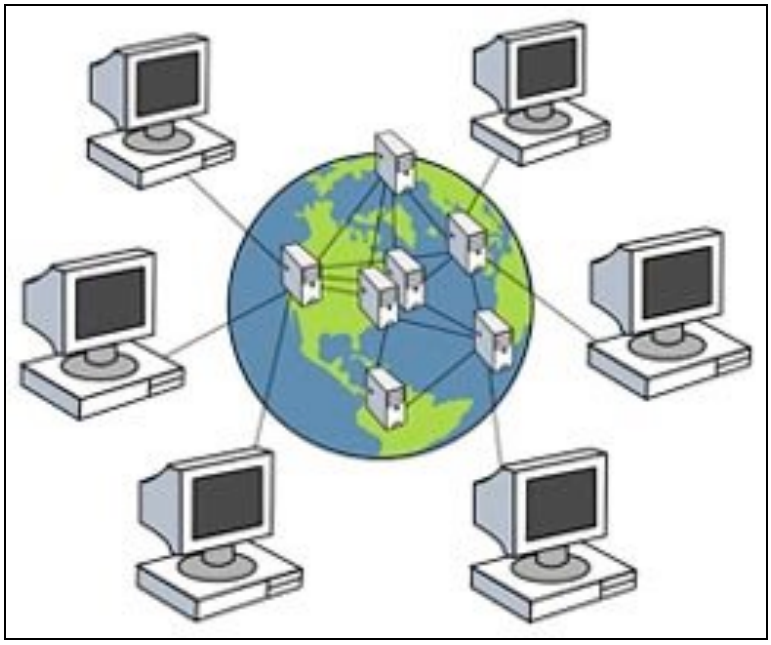


Gambar 2.4 Skema Jaringan MAN

2.2.1.3 *Wide Area Network* (WAN)

Wide Area Network (WAN) merupakan jaringan yang lebih besar dari MAN dan mencakup daerah geografis yang luas, seringkali mencakup sebuah negara atau benua. WAN terdiri dari kumpulan mesin yang bertujuan untuk menjalankan program-program (aplikasi) pemakai.

Pada sebagian besar WAN, jaringan terdiri dari sejumlah banyak kabel atau saluran telepon yang menghubungkan sepasang *router*. Bila dua *router* yang tidak menggunakan kabel yang sama akan melakukan komunikasi, keduanya harus berkomunikasi secara tidak langsung melalui *router* lainnya. Ketika sebuah paket dikirimkan dari sebuah router ke *router* lainnya melalui sebuah *router* perantara atau lebih, maka paket akan diterima *router* perantara dalam keadaan lengkap, disimpan sampai saluran *output* menjadi bebas, dan kemudian baru diteruskan (Tanenbaum, 1997). Skema jaringan WAN dapat dilihat pada Gambar 2.5.



Gambar 2.5 Skema Jaringan WAN (Komputer, Konsep Jaringan Komputer dan Pengembangannya, 2003)

2.2.1.4 *Client-Server*

Jenis jaringan ini terdapat komputer yang bertindak sebagai server dan komputer yang berperan sebagai *client* (*workstation*). Secara normal komputer server dapat mengontrol sepenuhnya komputer *client*. Server biasanya mempunyai kemampuan yang lebih besar dari pada *client* (Wahidin, 2008).

2.2.1.5 *Peer to Peer*

Peer to Peer merupakan jenis jaringan yang tidak memerlukan server secara khusus, karena komputer yang terhubung pada jaringan dapat bertindak sebagai server ataupun *client*. Contoh: hubungan komputer ke komputer (Wahidin, 2008).

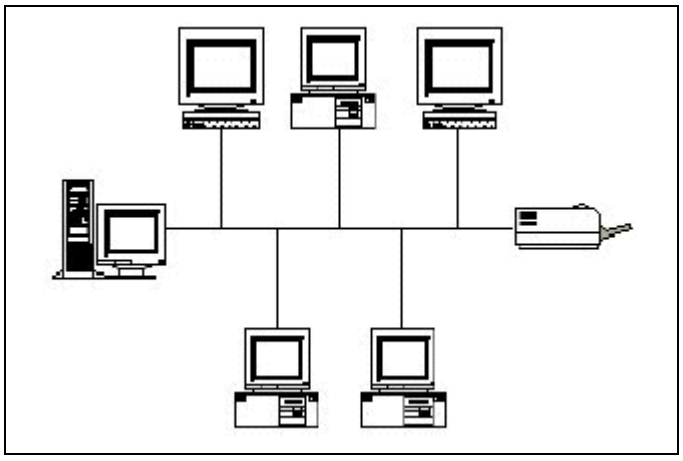
2.2.2 Topologi Jaringan Komputer

Topologi adalah suatu cara menghubungkan komputer yang satu dengan komputer lainnya sehingga membentuk jaringan. Masing-masing topologi ini mempunyai ciri khas, dengan kelebihan dan kekurangannya. Penulis akan

menjelaskan beberapa topologi jaringan, seperti: topologi *bus*, topologi *ring* dan topologi *star*.

2.2.2.1 Topologi *Bus*

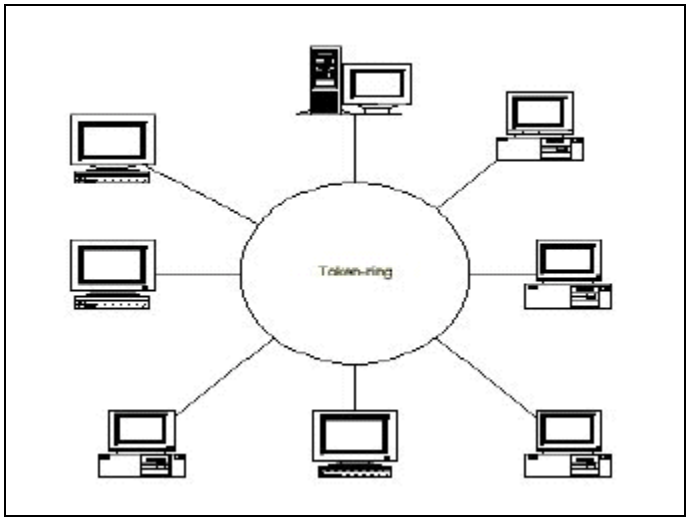
Layout ini termasuk *layout* umum. Satu kabel utama menghubungkan tiap simpul ke saluran tunggal komputer yang mengaksesnya ujung dengan ujung. Masing-masing simpul dihubungkan ke dua simpul lainnya, kecuali komputer di salah satu ujung kabel, yang masing-masing hanya terhubung ke satu simpul lainnya. Topologi ini seringkali dijumpai pada sistem *client*/server, dimana salah satu komputer pada jaringan tersebut difungsikan sebagai *file* server, yang berarti bahwa komputer tersebut dikhususkan hanya untuk pendistribusian data dan biasanya tidak digunakan untuk pemrosesan informasi. Dengan kata lain, pada topologi jenis ini semua terminal terhubung ke jalur komunikasi. Informasi yang akan dikirim akan melewati semua terminal pada jalur tersebut. Jika alamat yang tercantum dalam data atau informasi yang dikirim sesuai dengan alamat terminal yang dilewati, maka data atau informasi tersebut akan diterima dan diproses. Jika alamat tersebut tidak sesuai, maka informasi tersebut akan diabaikan oleh terminal yang dilewati. Skema topologi *bus* dapat dilihat pada Gambar 2.6. ((Komputer, Konsep Jaringan Komputer dan Pengembangannya, 2003)



Gambar 2.6 Skema Topologi *Bus* (Komputer, Konsep Jaringan Komputer dan Pengembangannya, 2003)

2.2.2.2 Topologi *Ring*

Topologi ini mirip dengan topologi *bus*, tetapi kedua terminal yang berada di ujung saling dihubungkan, sehingga menyerupai seperti lingkaran. Setiap informasi yang diperoleh diperiksa alamatnya oleh terminal yang dilewatinya. Jika bukan untuknya, informasi dilewatkan sampai menemukan alamat yang benar. Setiap terminal dalam jaringan saling tergantung, sehingga jika ada kerusakan pada satu terminal maka seluruh jaringan akan terganggu. Skema topologi *ring* dapat dilihat pada Gambar 2.7. (Komputer, Konsep Jaringan Komputer dan Pengembangannya, 2003)

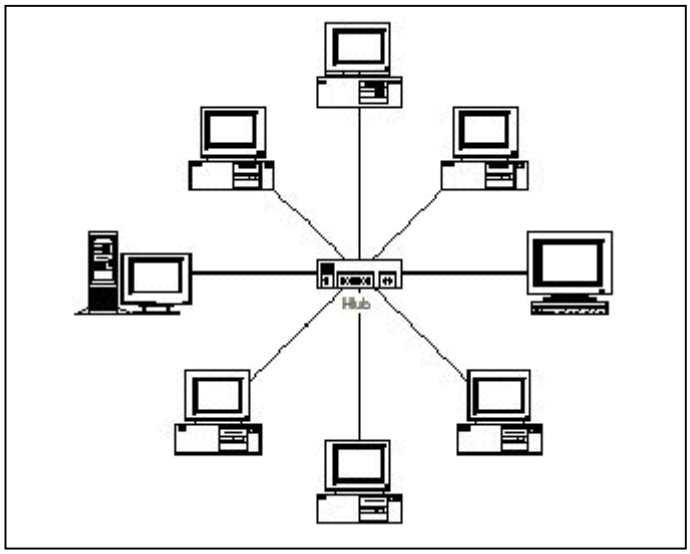


Gambar 2.7 Skema Topologi *Ring* (Komputer, Konsep Jaringan Komputer dan Pengembangannya, 2003)

2.2.2.3 Topologi *Star*

Dalam topologi *Star*, sebuah terminal pusat bertindak sebagai pengatur dan pengendali semua komunikasi data yang terjadi. Terminal-terminal lain terhubung padanya dan pengiriman data dari satu terminal ke terminal lainnya melalui terminal pusat. Terminal pusat menyediakan jalur komunikasi khusus untuk dua terminal yang akan berkomunikasi. Dengan kata lain semua kontrol dipusatkan pada satu komputer yang dinamakan stasiun primer dan komputer lainnya dinamakan stasiun sekunder. Setelah hubungan jaringan dimulai, setiap

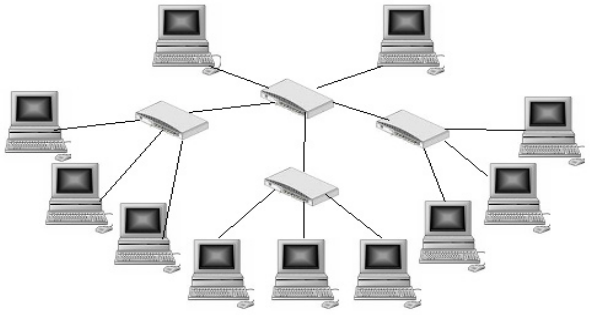
stasiun sekunder dapat sewaktu-waktu menggunakan hubungan jaringan tersebut tanpa menunggu perintah dari stasiun primer. Skema topologi *star* dapat dilihat pada Gambar 2.8. (Komputer, Konsep Jaringan Komputer dan Pengembangannya, 2003)



Gambar 2.8 Skema Topologi *Star* (Komputer, Konsep Jaringan Komputer dan Pengembangannya, 2003)

2.2.2.4 Topologi *Tree*

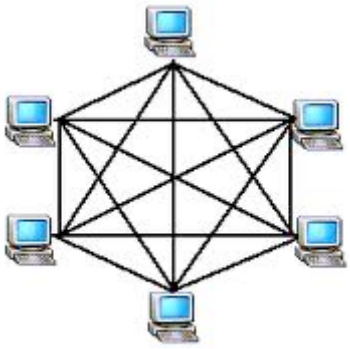
Topologi *tree* pada dasarnya merupakan perluasan dari topologi *star*. Seperti halnya Topologi *star*, perangkat *(node, device)* yang ada pada topologi *tree* juga terhubung kepada sebuah pusat pengendali (*central HUB/ Switch*) yang berfungsi mengatur *traffic* di dalam jaringan. Meskipun demikian, tidak semua perangkat pada topologi *tree* terhubung secara langsung ke *central HUB/Switch*. Sebagian perangkat memang terhubung secara langsung ke *central HUB/Switch*, tetapi sebagian lainnya terhubung melalui *secondary HUB/Switch*. Skema topologi *star* dapat dilihat pada Gambar 2.9. (Admin, 2014)



Gambar 2.9 Skema Topologi *Tree*

2.2.2.5 Topologi *Mesh*

Topologi *mesh* merupakan topologi dimana komputer dan perangkat jaringan saling terinterkoneksi satu dengan yang lainnya, mengijinkan untuk mendistribusikan tranmisi data meskipun jika salah satu koneksi terputus. Topologi ini jarang digunakan dalam jaringan saat ini karena sulit untuk diterapkan dan biaya yang mahal, namun topologi ini banyak digunakan untuk jaringan nirkabel. Skema topologi *mesh* dapat dilihat pada Gambar 2.10



Gambar 2.10 Skema Topologi *Mesh*

2.3 *Open Systems Interconnection* (OSI)

Open Systems Interconnection (OSI) adalah upaya standarisasi jaringan komputer yang dimulai pada tahun 1982 oleh *International Organization for Standardization* (ISO) bersama *International Telecommunication Union Telecommunication Standardization Sector* (ITU-T).

Sebelum OSI, sistem jaringan komputer sangat tergantung kepada pemasok (*vendor*). OSI berupaya membentuk standar umum jaringan komputer untuk menunjang interoperatibilitas antar pemasok yang berbeda. Dalam suatu jaringan yang besar biasanya terdapat banyak protokol jaringan yang berbeda. Tidak adanya suatu protokol yang sama, membuat banyak perangkat tidak bisa saling berkomunikasi.

Model OSI adalah salah satu kemajuan konsep ynag paling penting dalam jaringan komputer. Model ini membuat suatu ide model standar untuk lapisan protokol (*protocol layers*), dan mendefinisikan interoperatibilitas antara perangkat jaringan dan perangkat lunak.

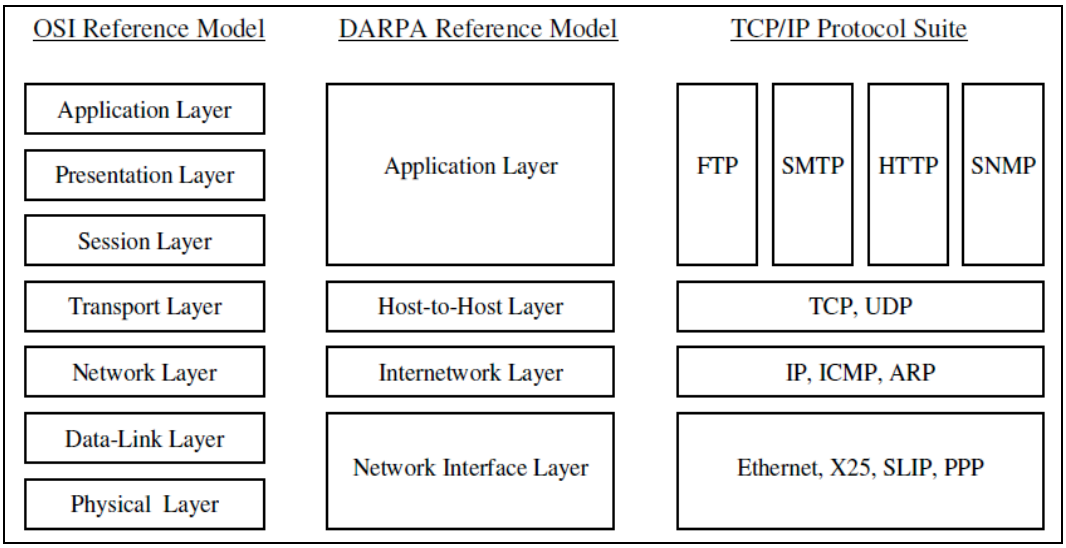
2.3.1 Model Referensi Jaringan Terbuka

Model referensi jaringan terbuka OSI atau *Reference Model for open networking* adalah sebuah model arsitektural yang dikembangkan oleh badan *International Organization for Standardization (ISO)* di Eropa pada tahun 1974. Model referensi ini pada awalnya ditujukan sebagai basis untuk mengembangkan protokol-protokol jaringan, meski pada kenyataannya inisiatif ini mengalami kegagalan. Kegagalan itu disebabkan oleh beberapa faktor berikut:

1. Standar model referensi ini, jika dibandingkan dengan model referensi DARPA (model internet) yang dikembangkan oleh *Internet Engineering Task Force* (IETF), sangat berdekatan. Model DARPA adalah model basis protokol TCP/IP yang populer digunakan.
2. Model referensi ini dianggap sangat kompleks. Beberapa fungsi (seperti halnya metode komunikasi *connectionless)* dianggap kurang bagus, sementara fungsi lainnya (seperti *flow control* dan koreksi kesalahan) diulang-ulang pada beberapa lapisan. Pertumbuhan internet dan protokol TCP/IP (sebuah protokol jaringan dunia nyata) membuat *OSI Reference Model* menjadi kurang diminati.

OSI Reference Model pun akhirnya dilihat sebagai sebuah model ideal dari koneksi logis yang harus terjadi agar komunikasi data dalam jaringan dapat berlangsung. Beberapa protokol yang digunakan dalam dunia nyata, semacam

TCP/IP, DECnet dan IBM *Systems Network Architecture* (SNA) memetakan tumpukan protokol *(protocol stack)* mereka ke OSI *Reference Model*. OSI *Reference Model* pun digunakan sebagai titik awal untuk mempelajari bagaimana beberapa protokol jaringan di dalam sebuah kumpulan protokol dapat berfungsi dan berinteraksi.



Gambar 2.11 Hubungan Antara *OSI Reference Model*

Pada Gambar 2.11 adalah hubungan antara *OSI Reference Model*, DARPA, dan protokol TCP/IP.

2.3.2 Konsep Dasar TCP/IP

Transmission Control Protocol/Internet Protocol (TCP/IP) adalah bukan sebuah protokol tunggal tetapi satu kesatuan protokol dan *utility*. Setiap protokol dalam kesatuan ini memiliki aturan yang spesifik. Protokol ini dikembangkan oleh ARPA (*Advance Research Project Agency*) untuk departemen pertahanan Amerika Serikat pada tahun 1969. ARPA menginginkan sebuah protokol yang memiliki karakter sebagai berikut:

1. Mampu menghubungkan berbagai jenis sistem operasi.
2. Dapat diandalkan dan mampu mendukung komunikasi kecepatan tinggi.

3. *Routabel* dan *scalable* untuk memenuhi jaringan yang kompleks dan luas. (Purbo, 1998)

2.3.3 Dasar Arsitektur TCP/IP

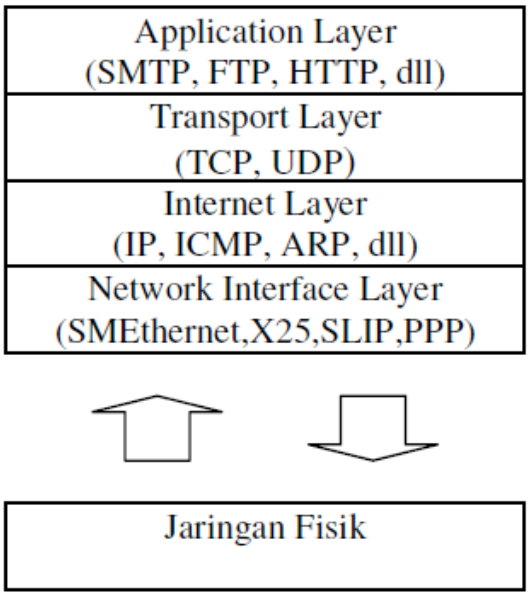
Pada dasarnya, komunikasi data merupakan proses mengirimkan data dari satu komputer ke komputer yang lain. Untuk dapat mengirimkan data, pada komputer harus ditambahkan alat khusus, yang dikenal sebagai *network interface* (*interface* jaringan). Jenis *interface* jaringan ini bermacam-macam, bergantung pada media fisik tersebut. Dalam proses pengiriman data ini terdapat beberapa masalah yang harus dipecahkan. Pertama, data harus dapat dikirimkan ke komputer yang tepat, sesuai tujuannya. Hal ini akan menjadi rumit jika komputer tujuan *transfer* data ini tidak berada pada jaringan lokal, melainkan di tempat yang jauh. Jika lokasi komputer yang saling berkomunikasi jauh (secara jaringan) maka terdapat kemungkinan data rusak atau hilang. Karenanya, perlu ada mekanisme yang mencegah rusaknya data ini. Hal lain yang perlu diperhatikan ialah, pada komputer tujuan *transfer* data mungkin terdapat lebih dari satu aplikasi yang menunggu datangnya data. Data yang dikim harus sampai ke aplikasi yang tepat, pada komputer yang tepat, tanpa kesalahan.

Untuk menangani semua masalah komunikasi data, keseluruhan aturan-aturan yang telah tersusun harus bekerja sama satu dengan yang lainnya. Sekumpulan aturan untuk mengatur proses pengiriman data ini disebut sebagai protokol komunikasi data. Protokol ini diimplementasikan dalam bentuk program komputer (*software*) yang terdapat pada komputer dan peralatan komunikasi data lainnya.

TCP/IP terdiri atas sekumpulan protokol yang masing-masing bertanggung jawab atas bagian-bagian tertentu dari komunikasi data. Atas prinsip ini, tugas masing-masing protokol menjadi jelas dan lebih sederhana. Protokol yang satu tidak perlu mengetahui cara kerja protokol yang lain, sepanjang ia masih bisa saling mengirim dan menerima data. (Purbo, 1998)

Karena penggunaan prinsip ini, TCP/IP menjadi protokol komunikasi data yang fleksibel. Protokol TCP/IP dapat diterapkan dengan mudah disetiap jenis

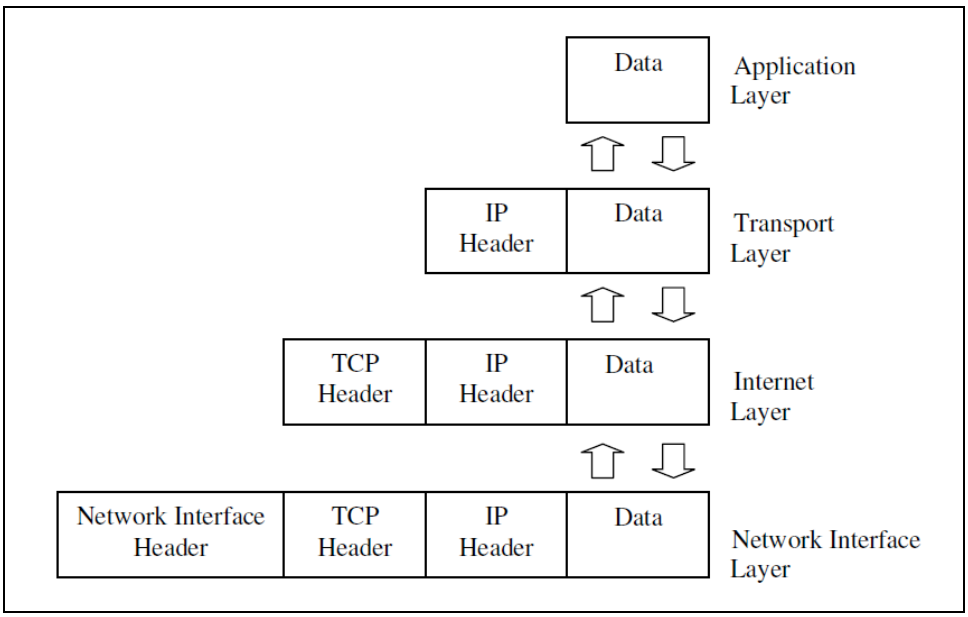
komputer dan *interface* jaringan, karena sebagian besar isi protokol ini tidak spesifik terhadap satu komputer atau peralatan jaringan tertentu. Agar TCP/IP dapat berjalan di atas jaringan *interface* jaringan tertentu, hanya perlu dilakukan perubahan pada protokol yang berhubungan dengan *interface* jaringan saja. Sekumpulan protokol TCP/IP ini dimodelkan dengan empat layer TCP/IP, sebagaimana terlihat pada Gambar 2.12 di bawah ini.



Gambar 2.12 *Layer TCP/IP*

Dalam TCP/IP, terjadi penyimpangan data dari protokol yang berada dalam satu *layer* ke protokol yang berada di *layer* lain. Setiap protokol memperlakukan semua informasi yang diterimanya pada protokol lain sebagai data. Jika suatu protokol menerima data dari protokol lain di layer atasnya, ia akan menambahkan informasi tambahan miliknya ke data tersebut. Informasi ini memiliki fungsi yang sesuai dengan protokol tersebut. Setelah itu, data ini akan diteruskan lagi ke protokol pada *layer* di bawahnya.

Hal yang sebaliknya terjadi jika suatu protokol menerima data dari protokol lain yang berada pada *layer* di bawahnya. Jika data ini dianggap *valid*, protokol akan melepas informasi tambahan tersebut, untuk kemudian meneruskan data itu ke protokol lain yang berbeda pada *layer* di atasnya. (Purbo, 1998)



Gambar 2.13 Pergerakan data dalam *layer TCP/IP*. (Tanenbaum, 1997)

TCP/IP terdiri atas 4 (empat) lapis kumpulan protokol yang bertingkat. Setiap lapisan yang dimiliki oleh *protocol suite* TCP/IP diasosiakan dengan protokolnya masing-masing. Protokol utama dalam protokol TCP/IP adalah sebagai berikut:

1. *Application Layer* : bertanggung jawab untuk menyediakan akses kepada aplikasi terhadap layanan jaringan TCP/IP. Protokol ini mencakup protokol *Dynamic Host Configuration Protocol* (DHCP), *Domain Name System* (DNS), *Hypertext Transfer Protocol* (HTTP), *File Transfer Protocol* (FTP), Telnet, *Simple Mail transfer Protocol* (SMTP), *Simple NetworkManagement Protocol* (SNMP), dan masih banyak protokol lainnya. Dalam beberapa inplementasi *stack* protokol, seperti halnya Microsoft TCP/IP, protokol-protokol lapisan aplikasi berinteraksi dengan menggunakan antarmuka *Windows Sockets* (*winsock*) atau NetBIOS over TCP/IP (NetBT).
2. *Transport Layer* : berguna untuk membuat komunikasi menggunakan sesi koneksi yang bersifat *connection-oriented* atau *broadcast* yang bersifat

connectionless. Protokol dalam lapian ini adalah *Transmission Control Protocol* (TCP) dan *User Datagram Protocol* (UDP).

3. Internet *Layer* : bertanggung jawab untuk melakukan pemetaan (*routing*) dan enkapsulasi paket-paket dan jaringan menjadi paket-paket IP. Protokol yang bekerja dalam lapisan ini adalah *Internet protocol* (IP), *Address Resolution Protocol* (ARP), *Internet Control Message protocol* (ICMP), dan *Internet Group Message Protocol* (IGMP).
4. *Network Interface Layer* : bertanggung jawab untuk meletakkan *frame-frame* jaringan di atas media jaringan yang digunakan. TCP/IP dapat bekerja dengan banyak teknologi *transport*, mulai dari teknologi *transport* dalam LAN (seperti halnya Ethertnet dan *Token Ring*).

2.3.4 *IP Address*

Dalam jaringan yang menggunakan protokol TCP/IP, setiap *host* akan memiliki alamat IP atau IP *address*. Format IP *address* adalah nilai biner berukuran 32 bit yang diberikan ke setiap *host* dalam jaringan. Nilai ini digunakan untuk mengenali jaringan dimana *host* tersebut dan mengenali nomor unik *host* bersangkutan di jaringan tertentu. Setiap *host* yang terhubung jadi satu pada sebuah internet *work* harus memiliki satu alamat unik TCP/IP. Konsep ini serupa dengan cara kantor pos mengantarkan surat. Setiap rumah di sepanjang jalan menggunakan nama jalan (nama jaringan) yang sama tetapi memiliki nomor rumah (nomor host) yang berbeda. Sewaktu-waktu komputer ingin mengirimkan data ke komputer lain, maka kiriman tersebut harus dilengkapi dengan alamat yang tepat. Jika tidak maka yang menerima atau jaringan akan kebingungan harus dikirim ke mana jaringan tersebut. Pemberian alamat ini menjadi tanggung jawab pengirim. Setiap alamat terbagi atas dua komponen, yaitu :

1. *Network ID*

   Network ID adalah bagian dari alamat IP yang mewakili jaringan fisik dari *host* (nama jalan rumah). Setiap komputer dalam segmen jaringan tertentu akan memiliki *ID* jaringan yang sama.

2. *Host ID*

   Host ID adalah bagian yang mewakili bagian individu dari alamat (nomor rumah). Bila komputer di segmen jaringan memiliki alamat, maka jaringan tersebut perlu tahu milik siapa paket itu. (Komputer, Konsep Jaringan Komputer dan Pengembangannya, 2003)

Jika dilihat dari bentuknya, IP *address* terdiri atas 4 buah bilangan biner 8 bit. Nilai terbesar dari bilangan biner 8 bit ialah 255 (=2+ 2+ 2+ 2+ 2+ 2+ 2+1). Karena IP address terdiri atas 4 buah bilangan 8 bit, maka jumlah IP *address* yang tersedia ialah 255 x 255 x 255 x 255.

IP *address* sebanyak ini harus dibagi-bagikan ke seluruh pengguna jaringan internet di seluruh dunia.

Untuk mempermudah proses pembagiannya, IP *address* dikelompokkan dalam kelas-kelas. Dasar pertimbangan pembagian IP *address* ke dalam kelas-kelas adalah untuk memudahkan pendistribusian pendaftaran IP *address*. Dengan memberikan sebuah ruang nomor jaringan (beberapa blok IP *address*) kepada ISP (*Internet Service Provider*) di suatu area diasumsikan penanganan komunitas lokal tersebut akan lebih baik, dibandingkan dengan jika setiap pemakai individual harus meminta IP address ke otoritas pusat, yaitu *Internet Assigned Numbers Authority* (IANA).

IP *address* ini dikelompokkan dalam lima kelas : Kelas A, Kelas B, Kelas C, Kelas D, dan Kelas E. Perbedaan pada tiap kelas tersebut adalah pada ukuran dan jumlahnya. IP kelas A dipakai oleh sedikit jaringan namun jaringan ini memliki anggota yang besar. Kelas C dipakai oleh banyak jaringan, namun anggota masing-masing jaringan sedikit. Kelas D dan E juga didefinisikan, tetapi tidak digunakan dalam penggunaan normal. Kelas D diperuntukkan bagi jaringan multicast, dan kelas E untuk keperluan eksperimental. (Tanenbaum, 1997)

Tabel 2.1 Kelas *IP Address*

Kelas	Rentang IP Address	Format Penulisan
A	1-126	1.0.0.0 s/d 126.255.255.255
B	128-191	128.0.0.0 s/d 191.255.255.255
C	192-223	192.0.0.0 s/d 223.255.255.255
D	224-239	224.0.0.0 s/d 239.255.255.255
E	240-255	240.0.0.0 s/d 254.255.255.255

2.4 *Virtual Local Area Networks* (VLAN)

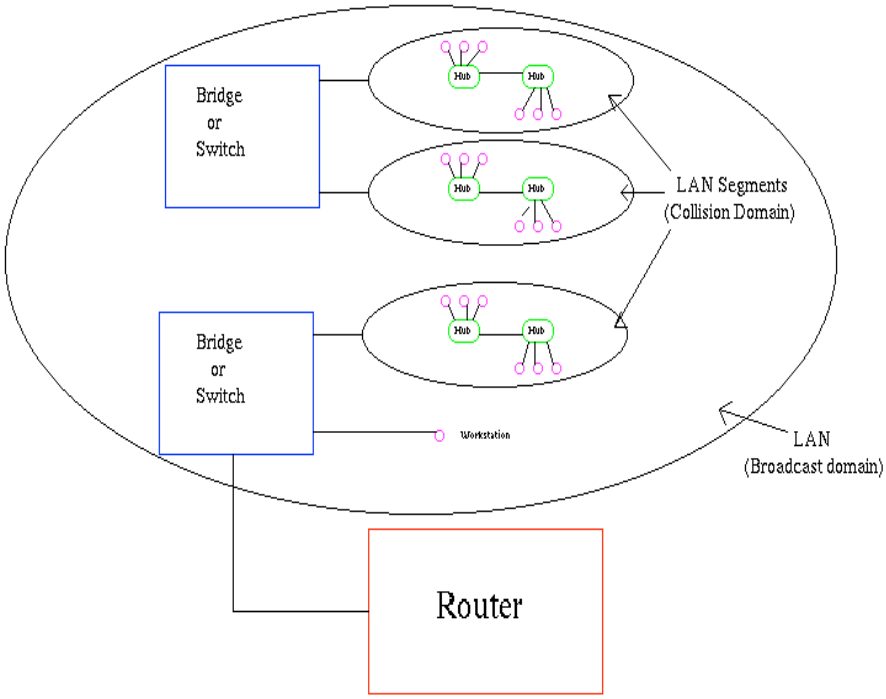
Jaringan LAN awalnya didefinisikan sebagai jaringan komputer yang terletak di daerah yang sama. Namun, pada dewasa ini jaringan LAN didefinisikan sebagai satu *broadcast domain*. Ini berarti bahwa jika satu pengguna memberikan atau menyebarkan informasi pada jaringan LAN nya, maka informasi yang disebarkan tersebut akan diterima oleh setiap pengguna LAN yang ada pada jaringan tersebut. Proses *broadcast* atau penyebaran informasi tersebut dapat dicegah dengan menggunakan *router*. Kerugian dari penggunaan metode ini adalah *router* biasanya mengambil lebih banyak waktu untuk memproses data yang masuk dibandingkan dengan penggunaan *switch* atau *bridge*. Hal yang lebih penting lagi yaitu, pembentukan *broadcast domain* tergantung pada koneksi fisik dari perangkat dalam jaringan. *Virtual Local Area Networks* (VLAN) dikembangkan sebagai solusi alternatif untuk penggunaan *router* yang berisi *broadcast traffic*.

2.4.1. Definisi VLAN

Dalam LAN konvensional *workstation-workstation* yang terhubungan satu sama lain dalam sebuah jaringan dihubungkan dengan sebuah hub atau *repeater*. Perangkat tersebut menyebarkan data yang masuk pada jaringan. Namun, jika dua *workstation* berusaha untuk mengirimkan informasi pada saat yang sama, maka tabrakan data akan terjadi dan semua data yang ditransmisikan akan hilang. Setelah tabrakan terjadi, hal tersebut akan terus disebarkan di seluruh jaringan dengan hub dan *repeater*. Oleh karena itu, informasi awal akan diperlukan untuk

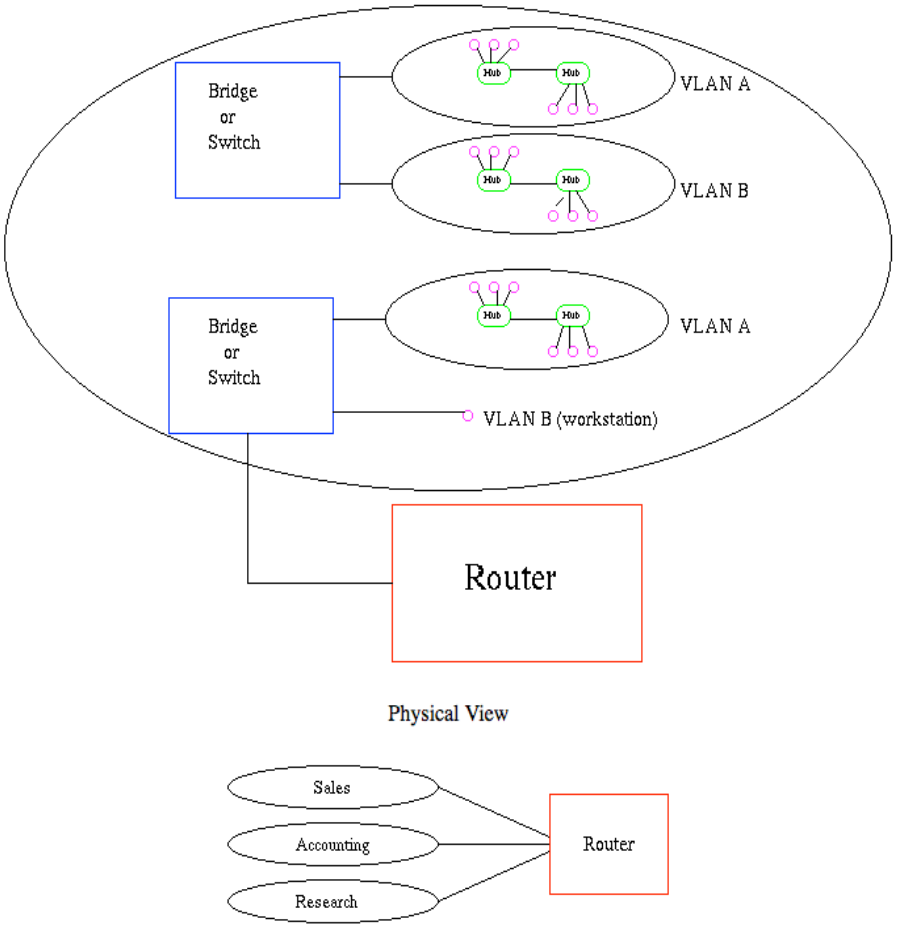
dikirimkan ulang setelah menunggu tabrakan data yang telah diselesaikan, sehingga hal tersebut akan menimbulkan pemborosan waktu dan sumber daya. Untuk mencegah hal tersebut diatas penggunaan sebuah *brigde* atau *switch* dapat digunakan untuk memperbaiki hal diatas. Perangkat ini tidak akan menimbulkan mem*forward collision*, tetapi hanya akan mengijinkan *broadcast* (untuk setiap pengguna di jaringan) dan *multicasts* (untuk *pre-spesified* sebuah grup para pengguna) untuk melewatinya. Sedangkan *router* memungkinkan untuk mencegah *broadcast* dan *multicast* dalam sebuah jaringan.

Workstation, hub, dan *repeater* secara bersama-sama membentuk sebuah segmen jaringan. Sebuah segmen LAN juga dikenal sebagai sebuah *collision domain* karena tabrakan atau *collision* yang terjadi tetap dalam segmen tersebut. Daerah di mana *broadcast* dan *multicast* terbatas disebut satu *broadcast domain* atau satu jaringan LAN. Jadi jaringan LAN dapat terdiri dari satu atau lebih segmen LAN. Hal ini mendefinisikan *broadcast* dan *collision domain* dalam sebuah LAN tergantung pada bagaimana *workstation*, hub, switch, dan router secara fisik terhubung bersama-sama. Hal ini mengartikan bahwa setiap orang yang terhubung ke jaringan harus pada area yang sama seperti yang ditunjukan pada .



Gambar 2.14 Skema Jaringan Fisik Untuk Jaringan LAN

Dengan VLAN memungkinkan manajer jaringan untuk logis segmen LAN ke dalam *domain broadcast* yang berbeda (lihat Gambar 2.15). Karena ini adalah segmentasi logis dan bukan satu fisik, *workstation* tidak harus secara fisik berada bersama-sama. Pengguna di lantai yang berbeda dari gedung yang sama, atau bahkan di bangunan yang berbeda kini dapat masuk ke LAN yang sama.



Gambar 2.15 Skema Fisik dan Lojik jaringan LAN

Selain itu juga VLAN mengijinkan *domain broadcast* tanpa penggunaan sebuah *router*. *Software bridging* digunakan sebaliknya untuk menentukan mana yang harus disertakan dalam *broadcast domain*. Router hanya akan digunakan untuk berkomunikasi antara dua VLAN. (Varadarajan, 1997)

2.4.2. Kelebihan VLAN

Beberapa kelebihan dari penggunaan VLAN dibandingkan LAN adalah sebagai berikut:

2.4.2.1. Performansi

Dalam sebuah jaringan yang mana mempunyai persentase *traffic* data yang tinggi baik itu *broadcast* dan *multicast*, maka dengan penggunaan VLAN ini dapat mengurangi kebutuhan untuk mengirimkan *traffic* data yang tidak perlu. Misalnya, dalam sebuah *broadcast domain* yang terdiri dari 10 *user*, jika *broadcast traffic* hanya untuk 5 *user*, kemudian memisahkan 5 *user* tersebut dengan VLAN yang terpisah maka akan dapat mengurangi *traffic*.

Dibandingkan dengan *switch*, *router* memerlukan proses yang lebih lama ketika *traffic* datang. Ketika *volume traffic* yang melewati *router* meningkat, begitu juga *latency* di *router* akan bertambah, yang mengakibatkan penurunan performansi. Penggunaan VLAN mengurangi jumlah *router* yang diperlukan, karena VLAN membuat *broadcast domain* dengan menggunakan *switch* bukan router. (Varadarajan, 1997)

2.4.2.2. Dapat membentuk Grup Virtual

Saat ini, itu adalah umum untuk menemukan tim pengembangan produk lintas fungsional dengan anggota dari berbagai departemen seperti pemasaran, penjualan, akuntansi, dan penelitian. Kelompok kerja ini biasanya dibentuk untuk waktu singkat. Selama periode ini, komunikasi antara anggota kelompok kerja akan tinggi. Untuk berisi siaran dan *multicast* dalam *workgroup*, VLAN dapat diatur untuk mereka. Dengan VLAN lebih mudah untuk menempatkan anggota kelompok kerja bersama-sama. Tanpa VLAN, satu-satunya cara yang ini akan mungkin untuk fisik pindah semua anggota *workgroup* lebih dekat bersama-sama.

Namun, kelompok kerja virtual tidak datang tanpa masalah. Pertimbangkan situasi di mana salah satu pengguna dari *workgroup* adalah di lantai keempat bangunan, dan anggota kelompok kerja lainnya berada di lantai dua. Sumber daya seperti printer akan terletak di lantai dua, yang akan nyaman untuk *lone* pengguna lantai empat.

Masalah lain dengan mendirikan kelompok kerja virtual adalah pelaksanaan *field server* terpusat, yang pada dasarnya koleksi *server* dan sumber daya utama untuk operasi jaringan di sebuah lokasi pusat. Keuntungan di sini

banyak, karena lebih efisien dan hemat biaya untuk menyediakan keamanan yang lebih baik, pasokan daya tak terputus, backup konsolidasi, dan lingkungan operasi yang tepat di daerah tunggal daripada jika sumber daya utama yang tersebar di sebuah bangunan. Peternakan *server* terpusat dapat menyebabkan masalah ketika mengatur virtual workgroups jika *server* tidak dapat ditempatkan pada lebih dari satu VLAN. Dalam kasus tersebut, *server* akan ditempatkan pada VLAN tunggal dan semua VLAN lain mencoba untuk mengakses *server* harus pergi melalui *router*; Hal ini dapat mengurangi kinerja. (Varadarajan, 1997)

2.4.2.3. Mempermudah Administrasi Jaringan

70% biaya jaringan adalah hasil dari pertambahan, pergerakan, dan perubahan dari *user* di dalam jaringan itu sendiri. setiap waktunya seorang *user* berpindah pindah dalam jaringan LAN, mengganti ulang kabel, alamat *station* yang baru, dan konfigurasi ulang hub dan *router* menjadi perlu. Beberapa tugas-tugas tersebut dapat disederhanakan dengan menggunakan VLAN. Jika seorang user bergerak atau berpindah dalam sebuah VLAN, konfigurasi ulang *router* menjadi tidak diperlukan. Selain itu, tergantung pada jenis VLAN, pekerjaan administratif lainnya dapat dikurangi atau dihilangkan. Namun kekuatan penuh dari penggunaan VLAN akan hanya benar-benar terasa saat alat manajemen yang baik diciptakan yang dapat memungkinkan manajer jaringan untuk *drag* dan *drop* ke *user* yang berbeda VLAN atau untuk mengatur alias.

Meskipun tabungan masuk secara didalam, VLAN akan menambahkan lapisan kompleksitas administratif, ketika saat ini menjadi penting untuk mengelola *workgroup* virtual. (Varadarajan, 1997)

2.4.2.4. Mengurangi Biaya

Vlan dapat digunakan untuk membuat *broadcast domain* yang mana mengeleminasi kebutuhan *router-router* mahal. (Varadarajan, 1997)

2.4.2.5. Keamanan

Secara berkala dan dalam suatu waktu, data yang bersifat sensitif dapat di *broadcast* dalam jaringan. Dalam kasus tersebut, menempatkan hanya para pengguna yang dapat memiliki akses ke data yang pada VLAN dapat mengurangi kemungkinan orang luar memperoleh akses ke data. VLAN juga dapat digunakan untuk mengontrol *broadcast domain*, mengatur *firewall*, membatasi akses, dan menginformasikan manajer mengenai penyusupan pada jaringan. (Varadarajan, 1997)

2.4.3. Cara Kerja VLAN

Pada saat sebuah *bridge* dalam jaringan LAN menerima data dari *workstation*, maka akan diberi tanda dengan *identifier* VLAN yang menunjukan dari mana data berasal, hal ini dinamakan *explicit tagging* (penandaan eksplisit). Selain itu juga memungkinkan untuk menentukan VLAN mana yang berasal dari data yang diterima dengan menggunakan *implicit tagging* (penandaan impilisit). Pada penandaan implisit, data tidak ditandai, melainkan VLAN dari mana data berasal dapat ditentukan berdasarkan *port* data berada. *Tagging* atau penandaan dapat berdasarkan pada *port* asal, misalnya dari *Media Access Control* (MAC), *network address*, atau beberapa *source* lainnya atau kombinasi dari beberapa source. VLAN diklasifikasikan berdasarkan pada metode yang digunakan. Untuk dapat melakukan *tagging* data yang menggunakan salah satu metode, *bridge* harus menjaga dan melakukan *update database* yang berisi *mapping* antara VLAN dan apapun yang digunakan untuk *tag*. Misalnya, jika pemberian *tag* oleh *port*, *database* harus menunjukkan port milik VLAN yang mana. Database ini disebut *filtering database*. *Bridge* akan mampu untuk menjaga *database* ini dan juga memastikan bahwa semua *bridge* di dalam jaringan LAN memiliki informasi yang sama di setiap *database*nya. *Bridge* menentukan kemana data akan pergi selanjutnya berdasarkan operasi normal LAN. Setelah *bridge* menentukan kemana data akan pergi, maka perlu untuk menentukan apakah VLAN *identifier* harus ditambahkan ke data dan dikirim. Jika data pergi ke perangkat yang mengetahui tentang implementasi VLAN (VLAN-*aware*), VLAN *identifier* ditambahkan ke

data. Jika data pergi ke perangkat yang tidak memiliki informasi mengenai implementasi VLAN (VLAN-*aware*), *bridge* mengirimkan data tanpa VLAN *identifier*.

Untuk memahami bagaimana VLAN bekerja, kita perlu melihat jenis VLAN, jenis koneksi antara *device* pada VLAN, *filtering database* yang digunakan untuk mengirimkan *traffic* ke VLAN yang benar, *tagging*, proses yang digunakan untuk mengidentifikasi VLAN dari mana berasal.

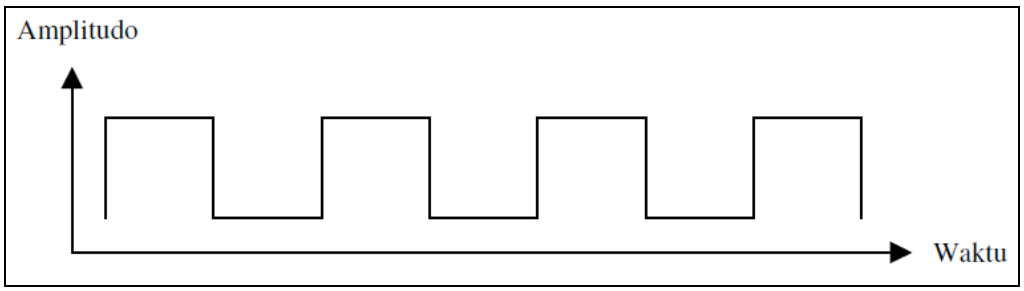
2.5 Sinyal Informasi

Data adalah suatu jenis informasi yang disimpan atau didapatkan kembali pada sebuah komputer. Oleh karena itu, jaringan mentransfer data dari satu komputer ke komputer yang lain. Data tersebut dapat berupa pesan e-*mail*, *file*, *web page*, video, musik dan lain sebagainya.

Sistem komunikasi jaringan pada komputer melambangkan data dengan menggunakan kode-kode yang diwakili secara efisien alat-alat elektronik dan gelombang radio. Sinyal tersebut membawa informasi melalui sistem dari satu titik ke titik yang lain. Sinyal tersebut dapat berupa sinyal digital atau analog.

2.5.1 Sinyal Digital

Sinyal pada komputer memiliki irama amplitudo yang berubah dari waktu ke waktu. Sinyal digital pada Gambar 2.10 biasanya berupa bilangan biner (dua digit), sehingga disebut dengan rangkaian digit biner (bit) atau data biner. Untaian digital dalam komputer dengan mudah menyimpan dan mengolah sinyal-sinyal digital tersebut ke dalam bentuk biner. (Geier, 2005)



Gambar 2.16 Sinyal Digital Ideal pada komputer

Bilangan biner merupakan sebuah sistem yang hanya menggunakan 0 dan 1 untuk merepresentasikan angka-angka. Konversi dari sistem bilangan desimal ke bilangan biner mudah dijalankan tinggal menyimpan bilangan biner tersebut.

Salah satu kelebihan sinyal digital adalah lebih mudah diperbaiki. Saat sinyal merambat melalui medium udara, sinyal tersebut dapat berbenturan dengan suara atau gelombang yang dapat merubah sinyal. Untuk mengatasi memperbaiki sinyal tersebut, maka untaian digital dapat mendeteksi jika ada denyut dalam periode waktu tertentu dan membuat denyut baru yang sama dengan denyut digital yang dikirim sebelumnya. Sinyal digital dapat menjangkau jarak jauh melalui periodik *repeater* sambil melindungi integritas informasi. (Geier, 2005)

Berikut ini hal-hal yang menetapkan karakteristik penting pada sinyal digital:

1. Kecepatan Data

Kecepatan data menyesuaikan dengan kecepatan yang ditransfer sinyal digital. Karena itu kecepatan data pada sinyal digital memberikan beberapa informasi mengenai lamanya pengiriman data dari satu titik ke titik yang lain dan mengidentifikasi jumlah *bandwidth* yang harus disuplai medium untuk mendukung sinyal secara efektif. Kecepatan data dari sinyal sama dengan waktu tempuh jumlah total bit yang ditransmisikan. Ukuran untuk kecepatan bit adalah *bits per second* (bps).

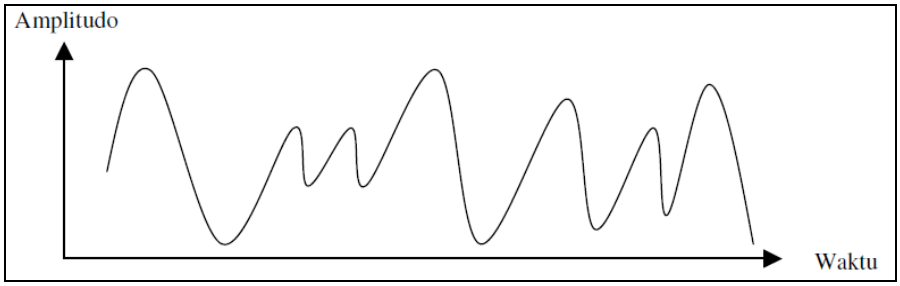
2. *Throughput*

Throughput sama dengan kecepatan data, akan tetapi kalkulasi *throughput* mengabaikan bit-bit yang berhubungan dengan *overhead* pada protokol komunikasi. Tidak ada standar untuk merepresentasikan *throughput*, kecuali jika

memasukkan informasi aktual yang dikirim menyebrangi jaringan. Karena itu, *throughput* memberikan solusi yang akurat untuk merepresentasikan performa dan efisiensi jaringan yang sebenarnya.

2.5.2 Sinyal Analog

Sinyal analog seperti ditunjukkan Gambar 2.13, merupakan salah satu amplitudo sinyal yang berubah secara terus-menerus dari waktu ke waktu.



Gambar 2.17 Sinyal Analog

Pada permulaan komunikasi elektronik, sebagian besar komunikasi elektronik mengolah sinyal dalam bentuk analog karena input informasinya berasal dari manusia. Sinyal analog memiliki amplitudo, *voltase*, energi, dan frekuensi. (Geier, 2005)

2.6 Bentuk Komunikasi

Pada penulisan ini, penulis akan menjelaskan dua jenis bentuk komunikasi, yaitu: *simplex* dan *duplex*.

2.6.1 Simplex

Simplex adalah salah satu bentuk komunikasi antara dua belah pihak, dimana sinyal-sinyal dikirim secara satu arah. Metode transmisi ini berbeda dengan metode *full-duplex* yang mampu mengirim sinyal dan menerima secara sekaligus dalam satu waktu atau *half-duplex* yang mampu mengirim sinyal dan

menerima sinyal meski tidak dalam satu waktu. Transmisi secara *simplex* terjadi di dalam beberapa teknologi komunikasi, seperti siaran televisi atau siaran radio.

Transmisi *simplex* tidak digunakan dalam komunikasi jaringan karena node-node dalam jaringan umumnya membutuhkan komunikasi secara dua arah. Memang, beberapa komunikasi dalam jaringan, seperti *video streaming*, terlihat seperti *simplex*, tapi sebenarnya lalu lintas komunikasi terjadi secara dua arah, apalagi jika protokol TCP yang digunakan protokol lapisan *transport*nya. (htt)

2.6.2 Duplex

Duplex adalah sebuah istilah dalam bidang telekomunikasi yang merujuk kepada komunikasi dua arah. Terdapat dua metode *duplexing*, yaitu:

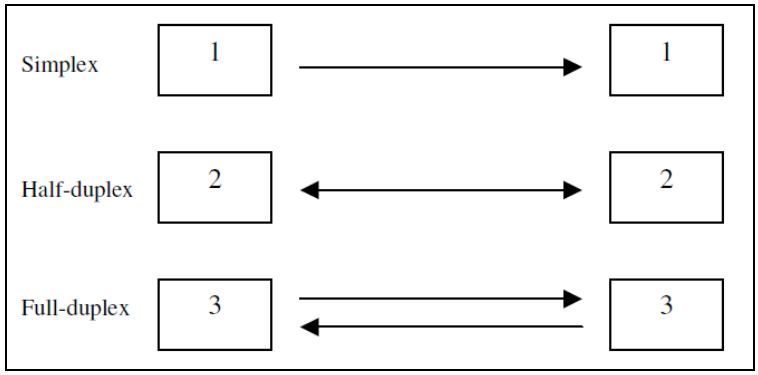
1. *Half duplex*

   Half duplex merupakan sebuah mode komunikasi dimana data dapat ditransmisikan atau diterima secara dua arah tapi tidak dapat secara bersama-sama. Contoh paling sederhana adalah *walkie-talkie*, dimana dua penggunanya harus menekan sebuah tombol untuk berbicara dan melepaskan tombol tersebut untuk mendengar. Ketika dua orang menggunakan *walkie-talkie* untuk berkomunikasi pada satu waktu tertentu, hanya salah satu diantara mereka yang dapat berbicara sementara pihak lainnya mendengar. Jika kedua-duanya mencoba untuk berbicara secara serentak, kondisi *collision* (tabrakan) pun terjadi dan kedua pengguna walkie-talkie tersebut tidak dapat saling mendengarkan apa yang keduanya kirimkan.

2. *Full duplex*

   Dalam komunikasi *full duplex*, dua pihak yang saling berkomunikasi akan mengirimkan informasi dan menerima informasi dalam waktu yang sama, dan umumnya membutuhkan dua jalur komunikasi. Komunikasi *full duplex* juga dapat diraih dengan menggunakan teknik *multiplexing*, dimana sinyal yang berjalan dengan arah yang berbeda akan diletakkan pada slot waktu (*time slot*) yang

berbeda. Kelemahan teknik ini adalah bahwa memotong kecepatan transmisi yang mungkin menjadi setengahnya .



Gambar 2.18 Cara Kerja Transmisi *Simplex, Half duplex,* dan *Full duplex*

2.7 Sejarah Sistem Operasi

Istilah sistem operasi sering ditujukan kepada semua *software* yang masuk dalam satu paket dengan sistem komputer sebelum aplikasi-aplikasi *software* terinstall. Dalam Ilmu komputer, sistem operasi atau dalam bahasa Inggris: *operating system* atau OS adalah perangkat lunak sistem yang bertugas untuk melakukan kontrol dan manajemen perangkat keras serta operasi-operasi dasar sistem, termasuk menjalankan *software* aplikasi seperti program-program pengolah kata dan *browser web*. Secara umum, sistem operasi adalah *software* pada lapisan pertama yang ditaruh pada memori komputer pada saat komputer dinyalakan. Sedangkan *software-software* lainnya dijalankan setelah sistem operasi berjalan, dan sistem operasi akan melakukan layanan inti umum untuk *software-software* itu. Layanan inti umum tersebut seperti akses ke *disk*, manajemen memori, *scheduling task*, dan antar-muka *user*. Sehingga masing-masing *software* tidak perlu lagi melakukan tugas-tugas inti umum tersebut, karena dapat dilayani dan dilakukan oleh sistem operasi.

2.7.1 Sejarah Linux

Linux pada awalnya dibuat oleh seorang mahasiswa Finlandia yang bernama Linus Benedict Torvalds. Dulunya Linux merupakan proyek hobi yang diinspirasikan dari Minix, yaitu sistem UNIX kecil yang dikembangkan oleh

Andrew Tanenbaum. Linux *versi* 0.01 dikerjakan sekitar bulan Agustus 1991. Kemudian pada tanggal 5 Oktober 1991, Linus mengumumkan versi resmi Linux, yaitu versi 0.02 yang hanya dapat menjalankan *shell bash(GNU Bourne Again Shell*) dan gcc (*GNU C Compile*r). (Komputer, Administrasi Jaringan Menggunakan Linux Ubuntu 7, 2008)

2.7.2 Distribusi Linux

Terdapat banyak distribusi Linux (lebih dikenali sebagai distro) yang dibuat oleh individu, grup, atau lembaga lain. Masing-masing disertakan dengan program sistem dan program aplikasi tambahan, di samping menyertakan suatu program yang memasang keseluruhan sistem di komputer (*installer program*). Inti di setiap distribusi Linux adalah kernel, koleksi program dari proyek GNU (atau proyek lain), cangkang (*shell*), dan aturcara utilitas seperti pustaka (*libraries*), kompilator, dan penyunting (*editor*). Kebanyakan sistem juga menyertakan aturcara dan utilitas yang bukan-GNU. Bagaimanapun, utilitas tersebut dapat dipisahkan dan sistem ala UNIX masih tersedia. Beberapa contoh adalah aturcara dan utiliti dari BSD dan sistem grafik-X (X-Window System). X menyediakan antarmuka grafis (GUI) yang umum untuk Linux. Contoh-contoh distribusi Linux : Debian, Ubuntu, SuSE, Fedora, Mandriva, Slackware, Debian, PCLinuxOS, Knoppix, Xandros. (Komputer, Administrasi Jaringan Menggunakan Linux Ubuntu 7, 2008)

2.7.3 Sejarah Windows

Microsoft Windows atau lebih dikenal dengan sebutan Windows adalah keluarga sistem operasi komputer pribadi yang dikembangkan oleh Microsoft yang menggunakan antarmuka dengan pengguna berbasis GUI (*Graphical User Interface*).

Dimulai dari DosShell for DOS 6 buatan Microsoft dan inginnya Microsoft bersaing terhadap larisnya penjualan Apple Macintosh yang menggunakan GUI, Microsoft menciptakan Windows 1.0. Nama ini berasal dari kelatahan karyawan Microsoft yang menyebut nama aplikasi tersebut sebagai

program windows (Jendela Program). Windows versi 2 adalah versi Windows pertama yang bisa diinstal program. Satu-satunya program yang bisa ditambahkan adalah Microsoft Word versi 1. Windows versi 3 menjanjikan aplikasi tambahan yang lebih banyak, kelengkapan penggunaan, kecantikan *user interface* atau antarmuka dan mudahnya konfigurasi.

Windows versi 3.1 adalah versi Windows yang bisa mengoptimalisasi penggunaannya pada prosesor 32-bit Intel 80386 ke atas. Windows versi 3.11 adalah versi Windows terakhir sebelum era *Start Menu*. Windows 3.11 pun adalah *versi* Windows pertama yang mendukung *networkin*g/jaringan. Versi Hibrida dapat dijalankan tanpa MS-DOS. Versi Hibrida tersebut menginstalasi dirinya sendiri dengan DOS 7. Tidak seperti Windows versi 16-bit yang merupakan *shell* yang harus diinstalasi melalui DOS terlebih dahulu. Aplikasinya pun berbeda. Meskipun Windows 9X dapat menjalankan aplikasi Windows 16-bit, namun Windows 9X memiliki *grade* aplikasi sendiri -X86-32, Windows 9X sangat terkenal dengan BSOD (*Blue Screen of Death*).

2.7.4 Versi – Versi Windows

Versi dari Windows sudah banyak yang telah diliris dari tahun 1985 sampai dengan tahun 2008, seperti penjelasan *point* dibawah ini.

16-bit, berjalan di atas MS-DOS

- 1985 November - Windows 1.0
- 1987 9 Desember - Windows 2.0
- 1990 22 Mei - Windows 3.0
- 1992 Agustus - Windows 3.1
- 1992 Oktober - Windows for *Workgroups* 3.1
- 1993 November - Windows for *Workgroups* 3.11

Hibrida (16-bit/32-bit), berjalan tanpa MS-DOS (meski tidak sepenuhnya)

- 1995 24 Agustus - Windows 95 (Versi: 4.00.950)
- 1998 25 Juni - Windows 98 (Versi: 4.1.1998)
- 1999 5 Mei - Windows 98 Second Edition (Versi: 4.1.2222)
- 2000 19 Juni - Windows *Millennium Edition* (Me) (Versi: 4.9.3000)

Berbasis kernel Windows NT

- 1993 Agustus - Windows NT 3.1
- 1994 September - Windows NT 3.5
- 1995 Juni - Windows NT 3.51
- 1996 29 Juli - Windows NT 4.0
- 2000 17 Februari - Windows 2000 (Versi: NT 5.0.2195)
- 2001 - Windows XP (Versi: NT 5.1.2600)
- 2003 - Windows Server 2003 (Versi: NT 5.2.3790)
- 2006 - Windows Vista (Versi 6.0 Build 6000)
- 2007 - Windows Home Server (Versi 6.0.1800.24)
- 2008 - Windows Server 2008 (Versi 6.1)
- 2009 - Windows 7 (Versi 6.1 Build 7600)
- 2009 - Windows Server 2008 R2 (Versi 6.1)
- 2011 - 2012 - Windows 8 (dalam perencanaan)

2.8 Manajemen Keamanan Informasi

Keamanan informasi terdiri dari perlindungan terhadap aspek-aspek berikut:

1. *Confidentiality* (kerahasiaan) : aspek yang menjamin kerahasiaan data atau informasi, memastikan bahwa informasi hanya dapat diakses oleh orang yang berwenang dan menjamin kerahasiaan data yang dikirim, diterima dan disimpan.
2. *Integrity* (integritas): aspek yang menjamin bahwa data tidak dirubah tanpa ada ijin pihak yang berwenang (*authorized*), menjaga keakuratan dan keutuhan informasi serta metode prosesnya untuk menjamin aspek integritas ini.
3. *Availability* (ketersediaan): aspek yang menjamin bahwa data akan tersedia saat dibutuhkan, memastikan user yang berhak dapat menggunakan informasi dan perangkat terkait (aset yang berhubungan bilamana diperlukan).

Keamanan informasi diperoleh dengan mengimplementasi seperangkat alat kontrol yang layak, yang dapat berupa kebijakan-kebijakan, praktek-praktek, prosedur-prosedur, struktur-struktur organisasi dan piranti lunak.

Terdapat beberapa langkah yang dapat dilakukan untuk pengendalian sistem keamanan untuk jaringan komunikasi dan informasi :

2.8.1 *Security Policy*

Security Policy (Kebijakan Keamanan), mengarahkan visi dan misi manajemen agar kontinuitas bisnis dapat dipertahankan dengan mengamankan dan menjaga integritas/keutuhan informasi-informasi krusial yang dapat dimiliki oleh perusahaan.

Security Policy sangat diperlukan mengingat banyak ditemuinya masalah-masalah non teknis salah satunya penggunaan *password* oleh lebih dari satu orang. Hal ini menunjukan tidak adanya kepatuhan dalam menerapkan sistem keamanan sistem informasi. Harus dilakukan inventarisasi data-data perusahaan. Selanjutnya dibuat peraturan yang melibatkan semua departemen sehingga peraturan dibuat peraturan yang melibatkan semua pihak. Setelah itu rancangan peraturan tersebut diajukan ke pihak direksi. Setelah disetujui, peraturan tersebut dapat diterapkan. *Security Policy* ini meliputi berbagai aspek,yaitu :

1. *Information security infrastructure*
2. *Information security policy*

2.8.2 *System Access Control*

System Access Control (Sistem Kontrol Akses), mengendalikan atau membatasi akses *user* terhadap informasi-informasi yang telah diatur kewenangannya termasuk pengendalian secara *tele-computing* ataupun *tele-networking.* Mengontrol tata cara akses terhadap informasi dan sumber daya yang ada meliputi beberapa aspek, yaitu :

1. *Access Control*
2. *User Access Management*
3. *User Responsibilities*

4. *Network Access Control*
5. *Operation System Access Control*
6. *Application Access Control*
7. *Monitor System Access and Use*
8. *Mobile Computing and Telenetworking*

2.8.3 *Communication and Operation Management*

Communication and Operation Management (Manajemen Komunikasi dan Operasi), menyediakan perlindungan terhadap infrastruktur sistem informasi melalui perawatan dan pemeriksaan berkala, serta memastikan ketersediaan panduan sistem yang terdokumentasi dan dikomunikasikan guna menghindari dan operasi yang terjadi meliputi berbagai aspek, yaitu :

1. *Operational Procedures And Responsibilities.*
2. *System Planning And Acceptance*
3. *Protection Against Malicious Software.*
4. *Housekeeping*
5. *Network Management*
6. *Media Handling And Security*
7. *Exchange Of Information And Software.*

2.8.4 *System Development and Management*

System Development and Management (pengembangan sistem dan pemeliharaan), memastikan bahwa sistem operasi maupun aplikasi yang baru diimplementasikan mampu bersinergi melalui verifikasi/validasi terlebih dahulu sebelum diluncurkan ke *live environement*.

1. *Security requirements of system*
2. *Security in application system*
3. *Cryptographic control*
4. *Security of system files*
5. *Security in development and support process.*

2.8.5 *Physical and Environmental Security*

Physical and Environmental Security (keamanan fisik dan lingkungan), membahas keamanan dari segi fisik dan lingkungan jaringan, untuk mencegah kehilangan dan kerusakan data yang diakibatkan oleh lingkungan, termasuk bencana alam dan pencurian data dalam media penyimpanan atau fasilitas informasi lain. Aspek yang dibahas antara lain :

1. *Secure Areas*
2. *Equipment Security*
3. *General Control*

2.8.6 *Compliance*

Compliance (penyesuaian), memastikan implementasi kebijakan-kebijakan keamanan selaras dengan peraturan dan perundangan yang berlaku, termasuk persyaratan kontraktual melalui audit sistem secara berkala. Kepatuhan yang mengarah kepada pembentukan prosedur dan aturan – aturan sesuai dengan hokum yang berlaku meliputi berbagai aspek, yaitu :

1. *Compliance with legal requirements*
2. *Reviews of security policy and technical compliance*
3. *System audit and consideration*

2.8.7 *Personal Security*

Personel security (keamanan perorangan), mengatur tentang pengurangan resiko dari penyalahgunaan fungsi penggunaan atau wewenang akibat kesalahan manusia (*human error*), sehingga mampu mengurangi kesalahan manusia dan manipulasi data dalam pengoperasian sistem serta aplikasi oleh *user*, melalui pelatihan-pelatihan mengenai *security awareness* agar setiap user mampu menjaga keamanan informasi dan data dalam lingkup kerja masing – masing. Aspek dari meliputi sebagai berikut ;

1. *Security In Job Definition And Resourcing*
2. *User Training*
3. *Responding To Security Incidens And Malfunction*

2.8.8 *Security Organization*

Security Organization (pengorganisasian keamanan), mengatur tentang keamanan secara global pada suatu organisasi atau instandi, mengatur dan menjaga ntegritas sistem informasi internal terhadap keperluan pihak eksternal

Termasuk pengendalian terhadap pengolahan informasi yang dilakukan oleh pihak ketiga (*outsourcing*). Aspek yang terlingkupi, yaitu :

1. *Security Of Third Party Access*
2. *Outsourcing*

2.8.9 *Asset Classification and Control*

Asset Classification and Control (pengklasifikasian dan pengontrolan aset), memberikan perlindungan terhadap asset perusahaan dan asset informasi berdasarkan *level* proteksi yang ditetntukan. Selain itu juga membahas tentang penjagaan asset yang ada meliputi berbagai aspek, diantaranya :

1. *Accountability For Assets*
2. *Information Classification*.

2.8.10 *Busines Continuity Management*

Busines Continuity Management (manajemen kelanjutan usaha), siap menghadapi resiko yang akan ditemui didalam aktivitas lingkungan bisnis yang bisa mengakibatkan "*major failure*" atau resiko kegagalan yang utama ataupun "*disaster"* atau kejadian buruk yang tak terduga, sehingga diperlukan pengaturan dan manajemen untuk kelangsungan proses bisnis.

2.9 *Network Access Control*

Network Access Control adalah sebuah jalan untuk keamanan jaringan komputer yang mencoba menyatukan teknologi keamanan *endpoint* (seperti antivirus, *host intrusion prevention*, dan *vulnaberity assessment*).

Network Access Control adalah solusi jaringan komputer yang menggunakan suatu kumpulan protokol untuk mendefinisikan dan mengimplementasikan sebuah kebijakan yang menjelaskan cara untuk

mengamankan akses ke suatu node jaringan oleh suatu perangkat ketika mereka mencoba untuk pertama kalinya. NAC mungkin saja menghubungkan proses *automatic remediation* (memperbaiki *node non-compliant* sebelum memperbolehkan untuk mengaskes) kedalam sistem jaringan, memperbolehkan infrastruktur jaringan seperti *router, switch* dan *firewall* untuk bekerja bersama dengan s*er*ver *back office* dan komputer *end user* untuk memastikan sistem informasi berjalan dengan aman sebelum kerja sama kedua sistem diperbolehkan.

Network Access Control memiliki tujuan yang sama dengan namanya yaitu mengkontrol akses ke jaringan dengan beberapa kebijakan, termasuk pemeriksaan sebelum dan *pasca* masuk untuk *endpoint*, dimana *user* dan perangkatnya dapat pergi ke suatu jaringan dan apa yang dapat mereka lakukan dalam jaringan tersebut.

Dengan kata lain ketika suatu komputer terhubung, komputer tersebut tidak diijinkan untuk mengakses apapun kecuali komputer tersebut mengikuti kebijakan yang telah didefinisikan; termasuk perlindungan anti-virus, *update* sistem, dan konfigurasi. Ketika komputer sedang dicek oleh *pre-installed software agent,* komputer hanya dapat mengakses sumber yang dapat memulihkan (memecahkan atau meng*update*) berbagai *issues*/masalah. Ketika kebijakan terpenuhi, komputer dapat mengakses sumber jaringan dan internet, dengan kebijakan yang telah ditentukan oleh NAC.

Banyak sekali solusi yang menawarkan NAC baik yang berupa *appliance* maupun berupa aplikasi yang diinstalkan pada server. Selain itu juga, ada yang berbayar ataupun yang gratis. Beberapa solusi NAC diantaranya adalah :

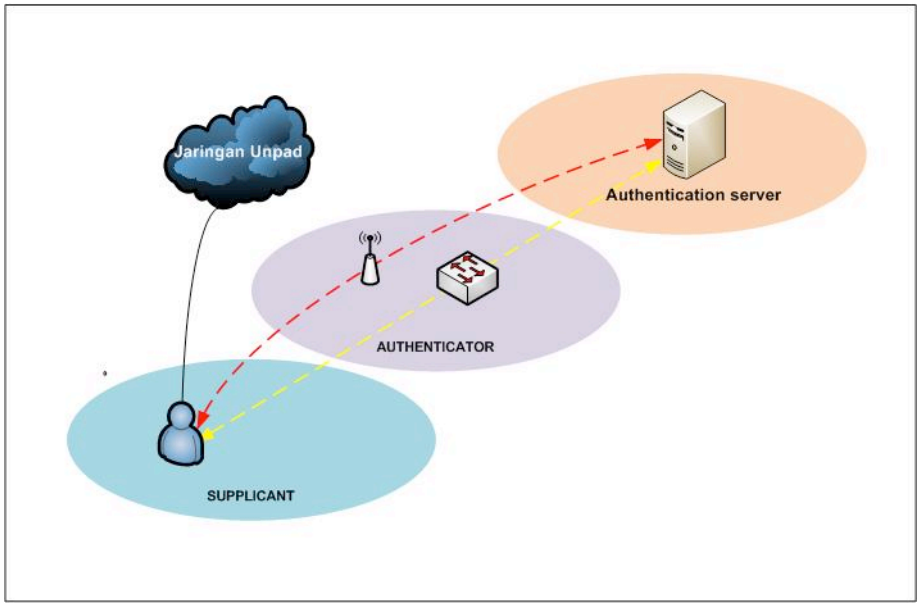
1. Cisco NAC Appliance.
2. ForeScout
3. Symantec NAC for security
4. McAfee NAC Appliance, Modul and Software
5. Extreme Network NAC Solution
6. Trustwave Network Access Control
7. Packetfence

Selain yang disebutkan diatas banyak lagi yang menawarkan solusi-solusi untuk *network access control* ini.

2.9.1. IEEE 802.1X

IEEE 802.1x merupakan protokol kontrol akses jaringan berbasis port yang memanfaatkan karakteristik infrastruktur LAN IEEE 802 untuk menyediakan media autentikasi dan autorisasi perangkat yang terhubung pada port LAN yang memiliki karakteristik koneksi *point to point,* dan mencegah akses jika autentikasi dan autorisasi gagal. Tujuan penggunaan standar ini IEEE 802.1x ini adalah untuk menghasilkan kontrol akses dan autentikasi terhadap perangkat user yang akan masuk ke jaringan Unpad.

IEEE 802.1x ini terdiri dari tiga bagian, yaitu pengguna atau user (mahasiswa, staff dan dosen Unpad) yang akan diauthentikasi atau yang disebut *supplicant,* server yang akan melakukan autentikasi yang disebut *Network Access Server* (*NAS*) *authentication server* yang digunakan adalah Remote Authentication Dial-In User Service (RADIUS) server dan digunakan untuk autentikasi pengguna yang akan mengakses wireless ataupun jaringan kabel. Yang ketiga adalah *authenticator* yang menghubungkan kedua bagian tersebut dalam hal ini switch manageable.



Gambar 2.19 Mekanisme *IEEE 802.1x*

2.10 Protokol Keamanan AAA

Menurut Jonathan Hassel (2002) konsep kerja *server* autentikasi dikenal dengan AAA *(authentication, authorization*, and *accounting).* Yang terdiri dari otentikasi, otorisasi, dan pendaftaran akun pengguna. Konsep AAA mempunyai fungsi yang berfokus pada tiga aspek dalam kontrol akses *user*, yaitu :

1. *Authentication*

Otentikasi adalah proses verifikasi untuk menyatakan suatu identitas. Bentuk umum yang biasa digunakan untuk melakukan otentikasi menggunakan kombinasi *login ID/username* dan *password*. jika kombinasi kedua nya benar maka *client* dapat mengakses ke sumber daya jaringan tertentu. Proses otentikasi dapat dianalogikan seperti seorang tamu yang datang ke rumah anda, sebelum tamu tersebut diperbolehkan masuk, tentu anda harus mengetahui tamu itu terlebih dahulu, jika anda kenal dengan tamu tersebut, maka tamu tersebut pastinya akan anda persilahkan masuk dan sebaliknya.

2. *Authorization*

Otorisasi melibatkan penggunaan seperangkat aturan - aturan yang berlaku untuk memutuskan aktifitas apa saja yang dizinkan dalam sistem atau sumber daya jaringan tertentu untuk pengguna yang terotentikasi. Proses Authorization merupakan lanjutan dari proses Authentication. Proses Authorization dapat dianalogikan sebagai berikut: jika anda sudah mengizinkan tamu untuk masuk kerumah anda, tentu anda mempunyai aturan – aturan yang ditempel di dinding rumah anda, misalnya tamu hanya boleh masuk sampai dengan ruang tamu. Dengan aturan seperti ini tentu akan memudahkan seseorang untuk melakukan kontrol terhadap sumber daya jaringan tertentu.

3. *Accounting*

Proses *accounting* merupakan proses dimana terdapat proses pencatatan berapa lama seorang pengguna sudah terkoneksi (waktu mulai / waktu stop) yang telah dilakukan selama pemakaian. Data dan informasi ini sangat berguna baik untuk pengguna maupun administrator, biasanya laporan ini digunakan untuk melakukan *auditing*, membuat laporan pemakaian, membaca karakteristik jaringan, dan pembuatan *billing* tagihan. Jadi pada intinya proses *accounting*

berguna untuk mengetahui apa saja yang dilakukan oleh *client* dan *service* apa saja yang dilakukan oleh *client*. Analogi sederhananya adalah mesin absensi di kantor, ia akan mencatat waktu datang dan waktu pulang, dengan demikian petugas dapat memonitoring karyawan dengan mudah.

2.10.1 *Remote Authentication Dial-In User Service* (RADIUS)

RADIUS merupakan singkatan dari *Remote Authentication Dial-In User Service*. Pertama kali dikembangkan oleh Livingston Enterprises. Merupakan *network* protokol keamanan komputer yang digunakan untuk membuat manajemen akses secara terkontrol pada sebuah jaringan yang besar. RADIUS didefinisikan dalam RFC 2865 dan RFC 2866. RADIUS biasa digunakan oleh perusahaan untuk mengatur akses ke internet bagi *client*.

RADIUS melakukan otentikasi, otorisasi, dan pendaftaran akun pengguna secara terpusat untuk mengakses sumber daya jaringan. Sehingga memastikan bahwa pengguna yang mengakses jaringan adalah pengguna yang sah. RADIUS berstandar IEEE 802.1x. Sering disebut "*port authentication based*". RADIUS merupakan protokol *client – server* yang berada pada layer aplikasi pada OSI layer. Dengan protokol *transport* UDP.(Jonathan Hassel, 2002).

2.10.1.1 Format Paket RADIUS

Protokol RADIUS menggunakan paket UDP untuk melewati transmisi antara *client* dan *server*. Protokol tersebut akan berkomunikasi pada *port* 1812. Berikut struktur paket RADIUS :



Gambar 2.20 Format Paket RADIUS

Format paket data RADIUS pada Gambar 2.19 terdiri dari 5 (lima) bagian, yaitu:

1. *Code* : memiliki panjang satu oktet, digunakan untuk membedakan tipe pesan RADIUS yang dikirimkan pada paket. Kode-kode tersebut (dalam desimal) dapat dilihat pada Tabel 2.2

Tabel 2.2 Kode Tipe Pesan RADIUS

Kode	Tipe pesan RADIUS
1	*Access-Request*
2	*Access-Accept*
3	*Access-Reject*
4	*Accounting-Request*
5	*Accounting-Response*
11	*Access-Challenge*
12	*Status-Server (experimental)*
13	*Status-Client (experimental)*
255	*Reserved*

2. *Identifier* : Memiliki panjang satu oktet, bertujuan untuk mencocokkan permintaan.
3. *Length* : Memiliki panjang dua oktet, memberikan informasi mengenai panjang paket.
4. *Authenticator* : Memiliki panjang 16 oktet, digunakan untuk membuktikan balasan dari RADIUS *server*, selain itu digunakan juga untuk algoritma *password*.
5. *Attributes* : Berisikan informasi yang dibawa pesan RADIUS, setiap pesan dapat membawa satu atau lebih atribut. Contoh atribut

RADIUS: nama pengguna, *password*, alamat IP *access point* (AP), pesan balasan. Tujuan standar 802.1x IEEE adalah untuk menghasilkan kontrol akses, autentikasi, dan manajemen kunci untuk *wireless* LAN. Standar ini berdasarkan pada *Internet Engineering Task Force* (IETF), *Extensible Authentication Protocol* (EAP), yang ditetapkan dalam RFC 2284.

2.10.1.2 Tipe Pesan Paket RADIUS

Ada 4 (empat) jenis paket pesan RADIUS yang relevan dengan otentikasi dan otorisasi pada fase transaksi AAA yaitu :

1. *Access-Request*

   Paket *Access-Request* digunakan oleh layanan konsumen ketika meminta layanan tertentu dari jaringan. *Client* mengirimkan paket *request* ke RADIUS *server* dengan daftar layanan yang diminta. Faktor kunci dalam transmisi ini adalah kolom kode pada *header* paket, dimana *header* paket tersebut harus di-*set* dengan nilai 1, yang merupakan nilai unik pada paket permintaan. RFC menyatakan bahwa balasan harus dikirimkan ke semua paket permintaan yang *valid*, apakah jawabannya adalah otorisasi atau penolakan.



Gambar 2.21 Paket *Access-Request*

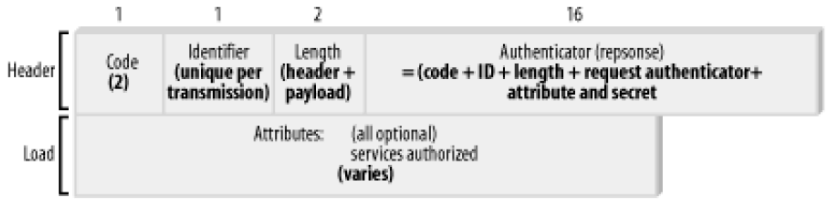
Tabel 2.3 Paket *Access-Request*

Identifier	Unique per request
Length	Panjang header ditambah semua atribut data tambahan
Authenticator	Request
Attribute Data	2 atau lebih

2. *Access-Accept*

Paket *Access-Accept* dikirim oleh RADIUS *server* kepada *client* untuk mengakui bahwa permintaan *client* diberikan. Jika semua permintaan Access-Request dapat diterima, maka server RADIUS harus mengatur paket respon dengan nilai 2 pada sisi *client*, setelah paket tersebut diterima, paket tersebut di cek apakah sama atau benar paket tersebut adalah paket respon dari RADIUS *server* dengan menggunakan *identifier field*. Jika terdapat paket yang tidak mengikuti standar ini maka paket tersebut akan dibuang.

Paket *Access-Accept* dapat berisi banyak atau sedikit atribut informasi yang perlu untuk dimasukkan. Kemungkinan besar atribut informasi pada paket ini akan menjelaskan jenis layanan apa saja yang telah dikonfirmasi dan resmi sehingga *client* dapat menggunakan layanan tersebut. Namun, jika tidak ada atribut informasi yang disertakan, maka *client* menganggap bahwa layanan yang diminta adalah yang diberikan.



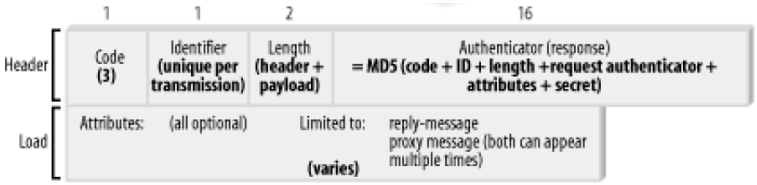
Gambar 2.22 *Paket Access-Accept*

Tabel 2.4 Paket *Access-Accept*

Packet Type	Response
Code	2
Identifier	Identical to Access-Request per transaction
Length	Header length plus all additional attribute data
Authenticator	Response
Attribute Data	0 or more

3. *Access-Reject*

RADIUS server dapat pula mengirimkan paket *Access-Reject* kembali ke *client* jika harus menolak salah satu layanan yang diminta *client* dalam paket *Access-Request*. Penolakan tersebut dapat didasarkan pada kebijakan sistem, hak istimewa yang tidak cukup, atau kriteria lain. *Access-Reject* dapat dikirim setiap saat selama waktu koneksi. Nilai yang diberikan untuk kode pada paket ini adalah 3.



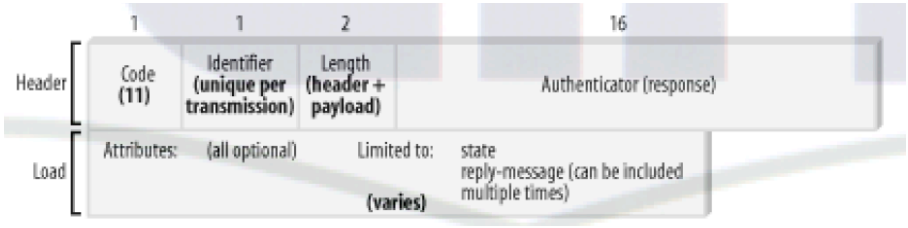
Gambar 2.23 Paket *Access-Reject*

Tabel 2.5 Paket *Access- Reject*

Packet Type	Response
Code	3
Identifier	Identical to Access-Request
Length	Header length plus all additional attribute data
Authenticator	Response
Attribute Data	0 or more

4. *Access-Challenge*

   Apabila *server* menerima informasi yang bertentangan dari *user*, atau membutuhkan informasi lebih lanjut, atau hanya ingin mengurangi risiko otentikasi palsu, *server* dapat menerbitkan paket *Access-Challenge* untuk *client*. Setelah client menerima paket *Access-Challenge* memberikan paket *Access-Request* yang baru disertai atribut informasi yang diminta *server*. Nilai yang diberikan pada *header* paket ini adalah 11.



Gambar 2.24 Paket *Access- Challenge*

Tabel 2.6 Paket *Access-Challenge*

Packet Type	Response
Code	11
Identifier	Identical to Access-Request
Length	Header length plus all additional attribute data
Authenticator	Response
Attribute Data	0 or more

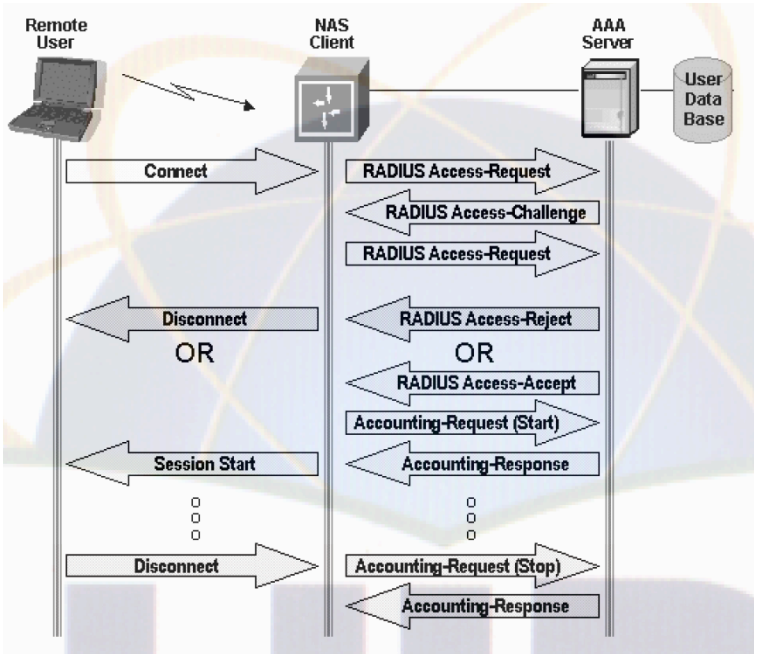
2.10.1.3 Tahapan Pesan RADIUS

Penggunaan RADIUS terhadap usaha pembentukan koneksi jaringan dapat dilakukan dengan melalui tahapan tahapan berikut:

1. *Access server*, *access point* menerima permintaan koneksi dari *access client*.
2. *Access server* dikonfigurasi agar dapat menggunakan RADIUS sebagai protokol yang melakukan proses otentikasi, otorisasi dan *accounting*,

membuat sebuah pesan *access request* dan mengirimkannya ke *server* RADIUS.

3. Server RADIUS melakukan evaluasi pesan *access request*
4. Jika dibutuhkan, server RADIUS mengirimkan pesan *access challenge* ke *access server*. Jawaban terhadap pesan tersebut dikirimkan dalam bentuk *access request* ke *server* RADIUS.
5. Surat kepercayaan dan otorisasi dari usaha pembentukan koneksi diverifikasi.
6. Jika usaha pembentukan koneksi dan diotorisasi, server RADIUS mengirimkan sebuah pesan *access accept* ke *access server*. Sebaliknya jika usaha tersebut ditolak maka server RADIUS mengirimkan sebuah pesan *access reject* ke *access server*.
7. Selama menerima pesan *access accept*, *access server* melengkapi proses koneksi dengan *access client* dan mengirimkan sebuah pesan *accounting request* ke server radius.
8. Setelah pesan *accounting request* diproses, *server* RADIUS mengirimkan sebuah pesan *accounting response*. (Zaenal Arifin, 2008).



Gambar 2.25 Proses Pembentukan Koneksi Protokol RADIUS

2.10.1.4 REALM

RADIUS hadir dengan kemampuan untuk mengidentifikasi *user* berdasarkan desain dan area yang berlainan atau disebut realm. Realm adalah *identifier* yang ditempatkan sebelum atau sesudah nilai yang biasanya berisikan atribut *username* yang bisa digunakan server RADIUS untuk mengenal dan menghubungi *server* yang sedang digunakan untuk memulai proses AAA.

Tipe pertama dari realm *identifier* yang dikenal sebagai realm prefix., yang mana nama realm ditempatkan sebelum *username*, dan kedua dipisahkan oleh karakter prekonfigurasi, seperti kebanyakan @, \, atau /. Untuk lebih lanjut, sebuah *user* bernama Jhassel yang terdaftar di layanan *central state* internet (merupakan realm dengan nama CSI) bisa mengatur *client* untuk memberikan *username* seperti CSI/jhassel.

Sintaks realm identifier lainnya adalah realm suffix, dimana *username* ditempatkan sebelum nama realm. Pemisah yang sama masih digunakan didalam sintaks ini hingga saat ini, lebih lanjut yang pada umum nya adalah tanda @. Sebagai contoh, *user* awatson mendaftar ke layanan *northwest* internet (nama realm : NWI) menggunakan identifikasi suffix realm bisa memberikan *username* seperti awatson@NWI.

2.11 Protokol Autentikasi

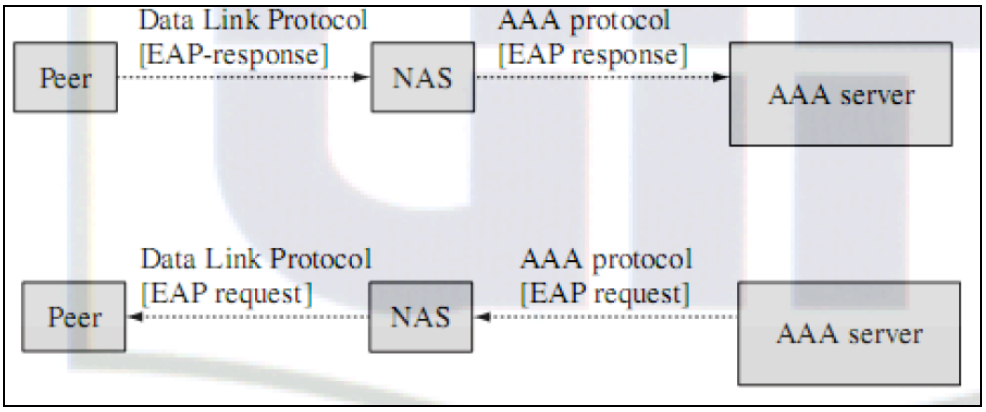
Protokol adalah suatu kumpulan dari aturan-aturan yang berhubungan dengan komunikasi data antara alat-alat komunikasi supaya komunikasi data dapat dilakukan dengan benar. Jabatan tangan merupakan contoh dari protokol antara dua manusia yang akan berkomunikasi. Pada istilah komputer, jabatan tangan (*handshaking*) menunjukkan suatu protokol dari komunikasi data bila dua buah alat dihubungkan satu dengan yang lainnya untuk menentukan bahwa keduanya telah kompatibel.

Otentikasi adalah proses verifikasi untuk menyatakan suatu identitas. Bentuk umum yang biasa digunakan untuk melakukan otentikasi menggunakan kombinasi *login ID/username* dan *password*, jika kombinasi kedua nya benar maka *client* dapat mengakses ke sumber daya jaringan tertentu. Proses otentikasi

dapat dianalogikan seperti seorang tamu yang datang ke rumah anda, sebelum tamu tersebut diperbolehkan masuk, tentu anda harus mengetahui tamu itu terlebih dahulu, jika anda kenal dengan tamu tersebut, maka tamu tersebut pastinya akan anda persilahkan masuk dan sebaliknya.

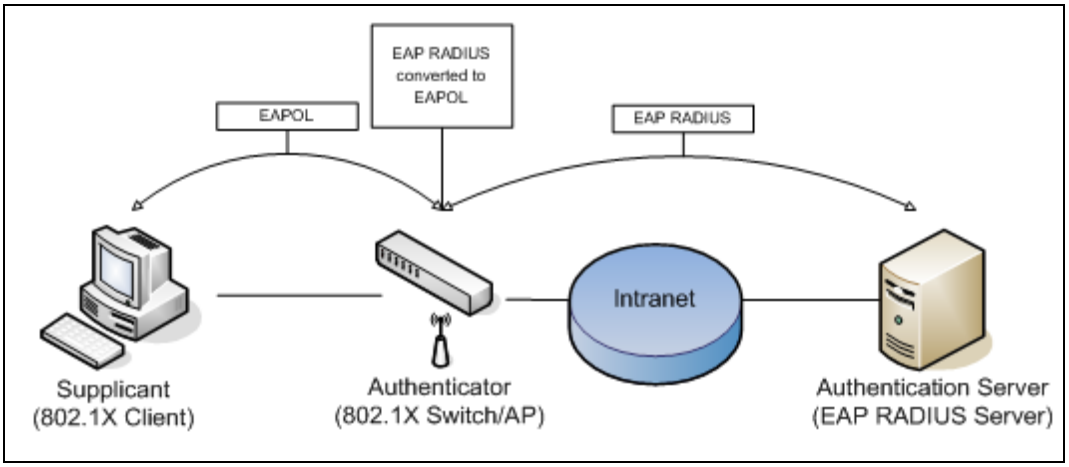
2.11.1 *Extensible Authentication Protocol* (EAP)

EAP atau *Extensible Authentication Protocol* adalah suatu *framework* otentikasi yang menyediakan layanan *transport* dan penggunaan *key* material dan parameter yang dihasilkan oleh EAP pada awalnya dikembangkan untuk koneksi *Point-to-Point* atau PPP. Namun, saat ini EAP juga diimplementasikan dan banyak digunakan untuk otentikasi penggunaan pada jaringan nirkabel. EAP digunakan pada *three-tier authentication*, maka pada proses komunikasinya EAP akan menggunakan *transport* protokol yang berbeda. Pertama, pada komunikasi antara *supplicant* dan *authenticator*, EAP akan menggunakan data *link* protokol seperti PPP, Ethernet atau WLAN. Kedua, pada komunikasi mengunakan *application layer* protokol seperti RADIUS atau Diameter.



Gambar 2.26 Proses Komunikasi Protokol EAP antara *Supplicant*, NAS dan *Authentication Server*

Dalam EAP terdapat beberapa komponen diantaranya :



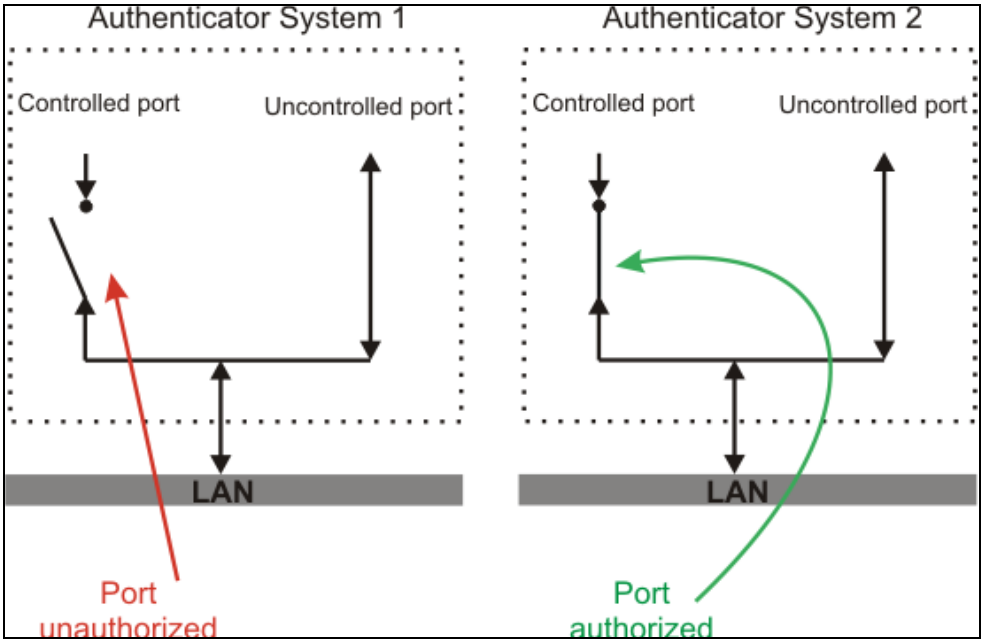
Gambar 2.27 Komponen EAP

1. *Supplicant*

   Merupakan *wireless node* yang ingin mengakses jaringan disebut *supplicant*.

2. *Authenticator*

   Merupakan perangkat yang memberikan akses menuju *server*. *Authenticator* merupakan *device* yang memproses apakah suatu *supplicant* dapat mengakses jaringan atau tidak. *Authenticator* mengontrol dua jenis *port* yaitu yang disebut dengan *controlled ports* dan yang disebut dengan *uncontrolled ports*. Kedua jenis *port* tersebut merupakan logikal *port* dan menggunakan koneksi fisikal yang sama. Sebelum otentikasi berhasil, hanya *port* dengan jenis *uncontrolled* yang dibuka. Trafik yang diperbolehkan hanyalah EAPOL atau EAPOW. Setelah *supplicant* melakukan autentifikasi dan berhasil, *port* jenis *controlled* dibuka sehingga *supplicant* dapat mengakses LAN secara biasa. IEEE 802.1x mempunyai peranan penting dari standar 802.11i.



Gambar 2.28 Skema *Port Based Authentication*

3. *Authentication Server* / RADIUS

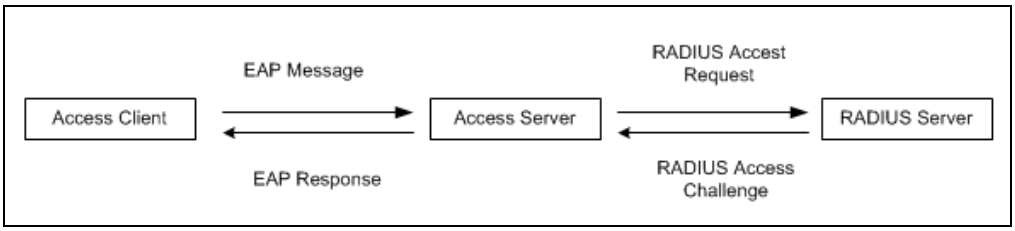
   Authentication Server / RADIUS adalah *server* yang menentukan apakah suatu *supplicant valid* atau tidak. *Authentication server* adalah berupa RADIUS *server* [RFC2865].

2.11.1.1 EAP *Over* RADIUS

EAP *over* RADIUS merupakan sebuah mekanisme otentikasi yang dilakukan oleh *access server (access point)* untuk melewatkan pesan EAP dari jenis EAP apa pun ke RADIUS *server* untuk melakukan proses otentikasi. Sebuah pesan EAP dikirim diantara *access client* dan *access server* dengan menggunakan format *attribute* EAP *Message* RADIUS dan dikirim sebagai pesan RADIUS antara *access server* dan *server* RADIUS. *Access server* hanya menjadi perangkat yang melewatkan pesan EAP diantara *client* dan *server* RADIUS. Pemrosesan pesan EAP dilakukan oleh *access client* dan *server* RADIUS.

EAP *over* RADIUS digunakan dalam lingkungan dimana RADIUS sebagai penyedia mekanisme otentikasi. Keuntungan yang bisa diperoleh dengan

menerapkan EAP *over* RADIUS adalah jenis EAP tidak perlu di-*install* pada setiap *access server*, cukup dilakukan pada *server* RADIUS. Tetapi, *access server* harus mendukung EAP sebagai protokol untuk melakukan kegiatan otentikasi dan melewatkan pesan EAP ke *server* RADIUS. Karena EAP merupakan bagian dari standar 802.1x, maka harus mengaktifkan otentikasi 802.1x untuk mengaktifkan AP agar dapat menggunakan EAP. Ketika koneksi dibuat, *access client* bernegosiasi dengan *access server* menggunakan EAP. Ketika *client* mengirimkan pesan EAP ke *access server*, *access server* membungkus pesan EAP dalam bentuk attribut EAP *message* kemudian diubah menjadi pesan RADIUS *access request* dan mengirimkannya ke *server* RADIUS. *Server* RADIUS memproses *attribute* EAP-*Message* dan mengirimkan sebuah pesan EAP-*Response* dalam bentuk pesan RADIUS *Access-Challenge* ke *access server*. Selanjutnya, *access server* melewatkan pesan EAP ke *access client*.



Gambar 2.29 Konversi Pesan EAP dan Pesan RADIUS

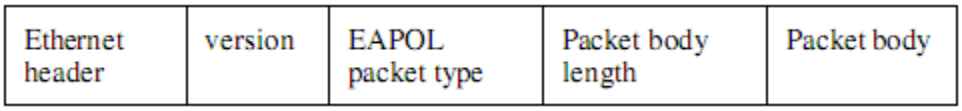
2.11.1.2 EAP Over LAN (EAPOL)

EAPOL adalah suatu protokol yang menyediakan cara-cara mengenkapsulasi paket-paket EAP dalam protokol LAN atau Ethernet. EAPOL didesain untuk standarisasi IEEE 802.1X yang digunakan pada jaringan kabel maupun nirkabel. Berikut ini adalah beberapa jenis paket-paket EAPOL, yaitu:

1. EAPOL-*Start* : merupakan sebuah *frame* EAPOL yang baru. *Frame* ini dikembangkan untuk mekanisme *network-sensing* pada jaringan nirkabel. Pada saat permintaan akses, *supplicant* akan mengirimkan *frame* EAPOL *start* secara *multicast* ke beberapa alamat MAC yang

telah dipersiapkan untuk 802.1x *authenticator*, sehingga *authenticator* dapat mengetahui apabila ada pengguna yang memerlukan ijin akses.

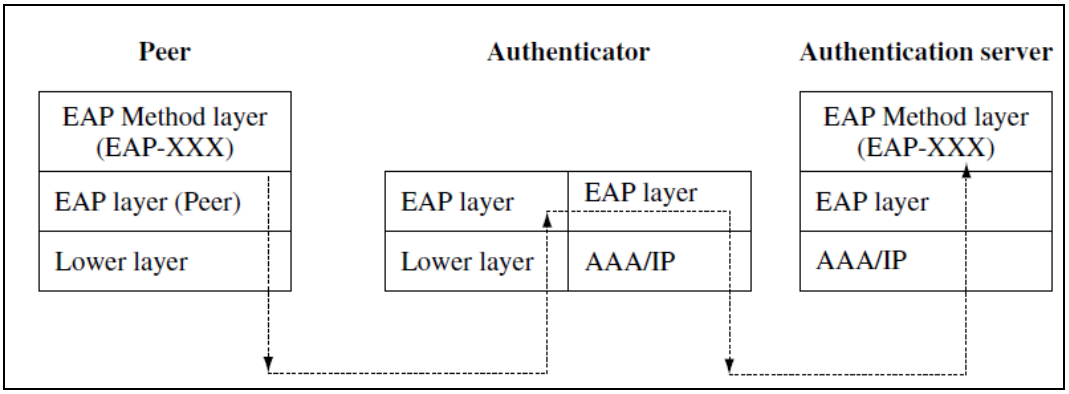
2. EAPOL-*Key* : *frame* ini dirancang untuk mendukung kombinasi antara proses otentikasi dan proses pendistribusian kunci, dimana *authenticator* akan mengirim kunci yang digunakan untuk enkripsi komunikasi data ke *supplicant*.
3. EAPOL-*Packet* : *frame* ini merupakan *frame* utama untuk protokol EAPOL, dimana *frame* ini bertugas untuk membawa paket atau pesan EAP. Semua tipe pesan EAP (EAP *request*, EAP *response*, EAP *success* dan EAP *failure*) dibawa oleh *frame* EAPOL-*Packet*.
4. EAPOL-*Logoff* : *frame* ini digunakan oleh *supplicant* untuk mengindikasikan bahwa pengguna ingin mengakhiri koneksi.



Gambar 2.30 Format paket EAPOL

2.12 EAP *Methods*

EAP sebenarnya hanya sebuah *authentication framework* dan tidak menyediakan mekanisme tertentu yang dapat digunakan untuk proses otentikasi. Tetapi, EAP menyediakan fungsi-fungsi umum dan negosiasi metode otentikasi yang disebut EAP *methods*. Beberapa EAP *methods* yang sering digunakan, yaitu EAP-MD5, EAP-OTP, EAP-GTC, EAP-SIM, EAP-AKA, EAP-TLS, EAP-TTLS dan PEAP. Ketiga EAP *methods* yang terakhir, yaitu EAP-TLS, EAP-TTLS dan PEAP adalah EAP *methods* yang sering digunakan pada jaringan nirkabel. EAP *methods* ini mendukung fitur mutual *authentication* dan penggunaan digital *certificate*. Pada EAP-TLS, *certificate digital* yang dibutuhkan ada dua, satu pada sisi *client* dan satu lagi pada sisi *server*, sedangkan pada EAP-TTLS dan PEAP, *digital certificate* pada sisi *client* bersifat *optional* dan dapat digantikan oleh kombinasi *username* dan *password*.



Gambar 2.31 Pemodelan Untuk Membawa Pesan Pada Otentikasi Dengan Metode TLS

2.12.1 EAP MD5

EAP-MD5 [RFC3748], merupakan IETF *open* standar tetapi menawarkan tingkat keamanan yang rendah. Fungsi *hash* MD5 mudah diserang dengan metode *dictionary attack*, tidak ada mutual otentikasi, dan penurunan kunci, sehingga membuatnya tidak cocok untuk dipakai dengan dinamik WEP atau WPA/WPA2 *enterprise*.

2.12.2 EAP TLS

EAP-TLS [RFC2716], adalah IETF standar dan banyak didukung oleh vendor peralatan *wireless*. EAP-TLS menawarkan tingkat keamanan yang tinggi, semenjak TLS dianggap sebagai teknik enkripsi yang sukses pada mekanisme SSL. TLS menggunakan *Public Key Infrastructure* (PKI) untuk mengamankan komunikasi antara *supplicant* dan *authentication server*. EAP-TLS adalah standar EAP *wireless* LAN. Meskipun EAP-TLS jarang digunakan, tetapi mekanismenya merupakan salah satu standar EAP yang paling aman dan secara *universal* didukung oleh semua manufaktur dari *wireless* LAN *hardware* dan *software* termasuk Microsoft.

2.12.3 EAP TLLS

EAP-*Tunneled* TLS atau EAP-TTLS merupakan standar yang dikembangkan oleh *Funk Software* dan *Certicom*. Standar ini secara luas di-*support* dan menawarkan tingkat keamanan yang bagus. Standar ini menggunakan PKI sertifikat hanya pada *authentication server*.

2.13 Packetfence

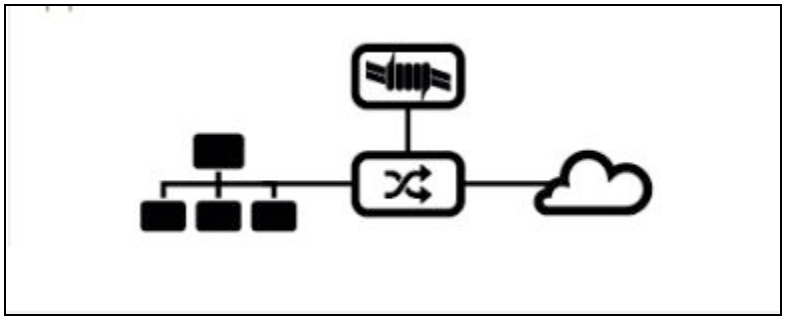
Packetfence merupakan aplikasi pendukung *open source* untuk solusi *network access control* (NAC). Packetfence mempunyai beberapa fitur diantaranya *captive-portal* untuk registrasi dan remediasi, *wireless management*, mendukung 802.1x, layer-2 *isolation* pada perangkat bermasalah, kemudian terintegrasi dengan *snort* IDS dan *nessus vulnerability scanner*. Packetfence dapat digunakan pada jaringan yang berskala kecil hingga jaringan yang berskala besar.

2.13.1 Arsitektur Packetfence

Dalam pengimplementasiannya packetfence mempunyai dua arsitektur yang dapat digunakan yaitu *inline deployment* dan *out-of-band deployment*.

2.13.1.1 *Out-Of-Band Deployment*

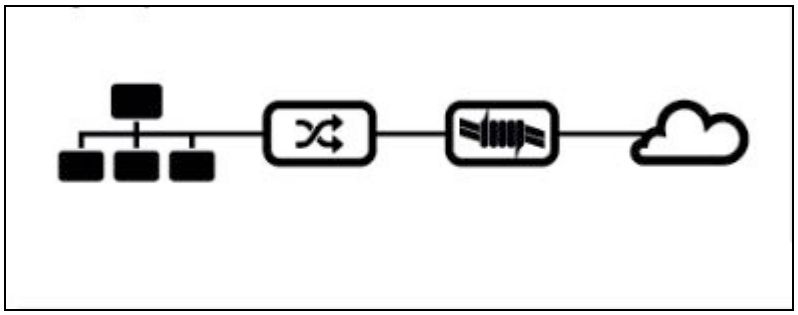
Arsitektur ini menempatkan *server* atau mesin packetfence diluar dari jalur koneksi *user*, artinya tidak dilewati secara langsung lalu lintas data sehingga lebih aman terhadap jaringan eksisting ketika terjadi kegagalan pada sistem packetfence itu sendiri. Arsitektur ini solusi untuk jaringan dengan skala yang besar.



Gambar 2.32 Arsitektur *Out-Of-Band Deployment*

2.13.1.2 *Inline Deployment*

Arsitektur ini menempatkan *server* packetfence berada pada jalur yang dilewati oleh lalu lintas data dari *user* menuju internet. Model arsitektur ini mendukung untuk jaringan berskala kecil dengan perang-perangkat yang tidak dapat di*manage.*



Gambar 2.33 Arsitektur *Inline Deployment*