KAJIAN PENGGUNAAN *PACKET FILTERING FIREWALL* MENGGUNAKAN *CISCO IP ACCESS CONTROL LIST*

Imam Sutoyo[1], Mochamad Wahyudi[2]
[1] Program Studi Teknik Komputer AMIK Bina Sarana Informatika
Jl. Kramat Raya No. 18 Jakarta Pusat, Indonesia
[2] Program Pascasarjana Magister Ilmu Komputer STMIK Nusa Mandiri
Jl. Salemba Raya No. 5 (10250) Jakarta Pusat, Indonesia
imam@bsi.ac.id
wahyudi@nusamandiri.ac.id

Abstract
A computer network has become a necessity for any organization implementing a computer-based information system. Hence, keeping the security aspect is important to maintain the network performance so as to provide optimum service to its users and to be up against any attacks especially when it is connected to the Internet. This paper is intended to give input to computer network administrators who implement IP Access Control List (ACL) network security system as firewall. It discusses the strengths and vulnerabilities of packet filtering firewall using Cisco IP Access Control List (ACL). The findings of this study will give computer network administrators better understanding on implementing Packet filtering firewall with Cisco IP ACL and comprehending the potential security holes due to its vulnerabilities.

Keywords : Vulnerabilities, Packet Filtering Firewall, Cisco IP Access Control List (ACL), Cracker

I. PENDAHULUAN

Untuk mengamankan suatu jaringan komputer, ada berbagai macam cara yang dapat dilakukan. *Firewall* istilah yang sangat dikenal dalam dunia keamanan jaringan, merupakan teknologi yang telah banyak diterapkan untuk melaksanakan fungsi keamanan jaringan.

Ada berbagai macam jenis *firewall*, namun *packet filtering* merupakan metode mendasar yang merupakan pondasi sebuah sistem *firewall.* Cisco *System* Inc. menyediakan fungsi *packet filtering* pada jajaran produk router mereka dengan nama *IP Access Control List* (ACL). ACL dapat digunakan sebagai *packet filtering firewall* yang bertugas melaksanakan fungsi penyaringan paket data yang akan masuk ke dalam jaringan internal *(inbound)* maupun yang akan keluar dari jaringan internal *(outboaund*).

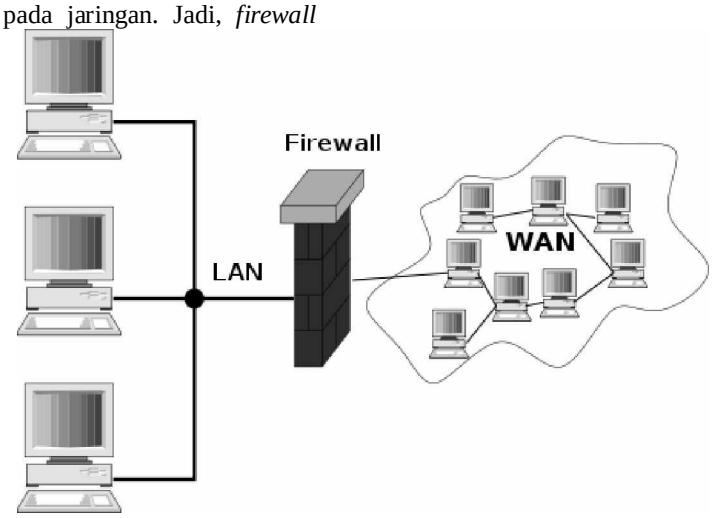
ACL merupakan suatu fasilitas keamanan yang terdapat pada Cisco *Internetwork Operating System* (IOS) yang terdapat pada suatu perangkat *router* keluaran Cisco. Jadi apabila kita akan menggunakan suatu perangkat *router* keluaran Cisco untuk membangun sebuah *internetwork*, kita tidak memerlukan kembali perangkat keamanan tambahan lain untuk membuat sebuah suatau sistem *firewall* sederhana.

ACL tentu memiliki banyak keterbatasan. Bagi para pengguna ACL, mengetahui keterbatasan atau kekurangan *(vulnerabilities)* ACL sangat penting. Hal ini dimaksudkan agar kita dapat diambil tindakan yang diperlukan untuk menutupi keterbatasan atau kekurangan *(vulnerabilities)* tersebut. Keterbatasan ACL, sebagaimana juga umumnya keterbatasan perangkat keamanan lainnya, tentu menimbulkan konsekuensi berupa *vulnerabilities* yang dapat berpotensi untuk dieksploitasi oleh *cracker*. Salah satu metode untuk menganalisa *vulnerabilitas* tersebut adalah dengan menggunakan metode *vulnerabilities taxonomi.*

II. PEMBAHASAN

2.1. *Firewall*

Firewall merupakan teknologi yang telah sangat dikenal dalam dunia keamanan jaringan. Menurut Brenton (2003) *firewall* adalah sebuah sistem atau kelompok sistem yang menerapkan sebuah kebijakan *access control* terhadap lalu lintas data yang melewati titik-titik akses pada jaringan. Jadi, *firewall* berfungsi untuk menyaring *(filtering)* lalu lintas data yang melewati titik-titik akses pada jaringan atau pintu-pintu keluar masuk lalu lintas data, baik yang ingin masuk ke dalam jaringan internal *(inbound)* maupun yang ingin keluar ke jaringan eksternal *(outbound)*.



Gambar 2.1. *Firewall*

Ada berbagai macam jenis *firewall*. Menurut Beny (2004), klasifikasi *firewall* menurut fungsi atau cara kerjanya adalah :

1. *Packet Filtering Firewall*

Sesuai dengan namanya, prinsip kerja f*irewall* jenis ini adalah melaksanakan penyaringan terhadap setiap paket data. Dari hasil penyaringan tersebut selanjutnya dapat diputuskan, apakah paket data tersebut dapat diproses lebih lanjut atau ditolak.

Firewall jenis ini umumnya diimplementasikan pada perangkat *router*. Router adalah perangkat jaringan yang bekerja pada lapisan jaringan *(network layer)* pada model *Open System Interconnection* (OSI). Dengan demikian, penyaringan paket data oleh *packet filtering firewall* setidaknya didasarkan pada informasi yang diolah pada lapisan tersebut, yaitu Alamat IP.

Kelebihan *firewall* jenis ini, antara lain : sifatnya independen, mudah disesuaikan dengan kebutuhan sistem, memiliki transparansi yang tinggi, dan unjuk kerjanya pun tinggi. Sebaliknya, kelemahan *firewall* jenis ini, antara lain pengamanan yang dilaksanakan masih sangat rendah dibandingkan dengan banyaknya ancaman yang mengintai jaringan komputer yang dijaga, sangat rentan terhadap *IP Spoofing*, tidak memiliki metode untuk memeriksa aktivitas yang dilakukan oleh koneksi-koneksi yang aktif *(stateless)*, tidak memiliki metode otentikasi, dan kemampuannya sangat terbatas karena hanya bekerja pada lapisan jaringan *(network layer)*.

2. Application Level Gateway

Firewall jenis ini biasa dikenal sebagai *proxy*. Prinsip kerjanya adalah sebagai perantara antara *host* pada jaringan internal dengan sumber daya eksternal yang diakses oleh *host* tersebut. Nama yang umum dikenal untuk *firewall* jenis ini adalah *proxy server.*

Dengan menggunakan *proxy server, host* pada jaringan internal tidak pernah berhubungan lansung dengan sumber daya jaringan di luar jaringan lokal tempat ia berada. Setiap ada permintaan koneksi untuk mengakses sumber daya jaringan di luar jaringan lokal harus selalu diarahkan ke *proxy server* terlebih dahulu. *Proxy server* inilah yang nantinya akan memutuskan, apakah

koneksi boleh dilaksanakan atau tidak.

Kelebihan f*irewall* jenis ini, antara lain pengamanan yang dilaksanakan lebih bagus dibandingkan *packet filter*, memiliki metode otentikasi, memiliki metode kendali akses *(access control)*, memiliki fasilitas *logging*, dan memiliki fasilitas *caching* untuk membantu menghemat *bandwith*. Sebaliknya, kelemahan *firewall* jenis ini, antara lain : kurangnya transparansi terhadap pengguna dimana aplikasi pengguna harus dikonfigurasi untuk mendukung fungsi *proxy*, aplikasi yang digunakan harus mendukung fasilitas *proxy*, dan unjuk kerja yang lebih rendah dibandingkan *packet filter*.

3. Circuit Level Gateway

Firewall jenis ini merupakan pengembangan dari *Application Level Gateway*. Prinsip kerja *Circuit Level Gateway* serupa dengan *Application Level Gateway*, yakni sebagai perantara antara *host* pada jaringan internal dengan sumber daya eksternal yang diakses oleh *host* tersebut. Perbedaaannya adalah pada tingkatan atau lokasi pelaksanaan fungsi perantaraan *(proxy)* tersebut dilaksanakan.

Kelebihan f*irewall* jenis ini kurang lebih sama dengan *Application Level Gateway*, namun ia menutupi kelemahan kurangnya transparansi dari *Application Level Gateway*, dimana setiap aplikasi tidak perlu dikonfigurasi untuk mendukung fungsi *proxy*, bahkan aplikasi yang tidak memiliki dukungan terhadap fungsi *proxy* pun masih dapat berjalan. Sebaliknya, kelemahan *Firewall* jenis ini pun kurang lebih sama, bedanya adalah aplikasi harus kompatibel dengan *platform* yang digunakan, misalnya harus kompatibel dengan *Application Programming Interface* (API).

4. Statefull Inspection

Firewall jenis ini adalah *firewall* yang paling canggih dibandingkan dengan tiga jenis *firewall* sebelumnya. Prinsip kerja dari *Statefull Inspection* adalah selalu aktif mengawasi setiap koneksi yang terjadi, sehingga selalu dapat diketahui status dari koneksi-koneksi tersebut *(statefull)* dan dapat dilaksanakan tindakan yang semestinya jika ditemukan adanya penyimpangan-penyimpangan dari koneksi yang ada.

Kelebihan *firewall* jenis ini, antara lain tingkat pengamanannya paling tinggi, pengamanannya paling lengkap karena mendukung dan dapat melaksanakan pengawasan pada seluruh lapisan OSI, memiliki unjuk kerja yang tinggi, memiliki *skalabilitas* yang bagus, dan memiliki transparansi yang tinggi. Sebaliknya, kelemahan *firewall* jenis ini adalah diperlukannya sumber daya yang sangat besar untuk menjalankannya, apalagi saat jumlah koneksi makin bertambah banyak.

2.2. Cisco *IP Access Control List* (ACL)

Pengendalian akses *(access control)* merupakan mekanisme pengamanan yang umum diimplementasikan dalam skenario pengamanan sebuah sistem informasi. *Access Control* dapat diterapkan melalui ketiga komponen sistem informasi, yaitu *hardware*, *software*, dan *brainware*.

Salah satu contoh penerapan *Access Control* melalui *hardware* secara *embedded*, lebih spesifik lagi pada sebuah perangkat *router* adalah seperti yang diterapkan oleh Cisco Systems Inc. Pada jajaran produk router mereka, yaitu Cisco *IP Access Control List* (ACL). ACL merupakan sebuah fasilitas keamanan yang dimiliki oleh perangkat *router* Cisco untuk menyaring paket data yang masuk maupun yang keluar dari *router*.

Menurut Cisco (2008), "*IP Access Control List* adalah sebuah daftar berurut yang paling sedikit terdiri dari satu pernyataan *permit* dan mungkin satu atau lebih pernyataan *deny*". Mekanisme penyaringan paket data pada ACL didasarkan pada dua pernyataan tersebut, yakni sebuah paket data akan diteruskan jika memenuhi kriteria pada pernyataan *permit* dan tidak memenuhi kriteria pada pernyataan *deny* dan sebaliknya paket data tidak akan diteruskan jika tidak memenuhi kriteria pada pernyataan *permit* atau memenuhi kriteria pernyataan *deny*. Terurut maknanya adalah, daftar pernyataan pada ACL diproses secara terurut atau *sekuensial* dari baris pertama hingga baris terakhir.

2.2.1. Klasifikasi ACL

Ada dua jenis ACL, yaitu *Standard ACL* dan *Extended ACL*. Sesuai dengan nama yang diberikan oleh pembuatnya, *Extended ACL* memiliki fasilitas *filtering* yang lebih lengkap dibandingkan *Standard ACL*, namun dalam

penggunaannya keduanya saling melengkapi.

Cisco memberikan arahan berkaitan dengan penempatan ACL, bahwa *Standard ACL* efektif jika diletakkan dilokasi tujuan, sedangkan *Extended ACL* seharusnya diletakkan pada lokasi asal dari paket data.

1. Standard ACL

Standard ACL hanya menyaring paket data berdasarkan alamat IP pengirim paket. Saat sebuah paket data ingin melewati sebuah *interface* dari *router*, alamat pengirim akan dibandingkan dengan alamat IP yang didefinisikan pada baris-baris pernyataan yang terdapat pada ACL yang dipasang pada *interface* tersebut. Paket data akan dilewatkan atau tidak berdasarkan hasil perbandingan.

Standard ACL menggunakan penomoran dari 1 sampai 99. Berikut ini sintaks dari *Standard ACL*.

access-list access-list-number {deny | permit} source-ip-address [wildcard-mask]

Dimana :

a. *Access-list*
   Keyword yang digunakan untuk membuat ACL.
b. *access-list-number*
   Nomor yang menjadi identitas dari ACL, jangkauannya adalah 1-99.
c. *{permit | deny}*
   Tindakan terhadap paket data. Paket data dapat dilewatkan atau ditolak.
d. *source-ip-address*
   Alamat IP pengirim.
e. *[wildcard-mask]*
   Wildcard yang digunakan.

Contoh :

access-list 1 permit host 202.101.51.3 0.0.0.0

Standard ACL tersebut hanya akan meneruskan paket dari alamat IP 202.101.51.3.

2. *Extended ACL*

Extended ACL memiliki fasilitas *filtering* yang lebih lengkap dibandingkan *Standard ACL*. Selain penyaringan paket data berdasarkan alamat IP pengirim, *Extended ACL* dapat digunakan untuk menyaring paket data berdasarkan alamat IP tujuan, nomor *port*, dan jenis protokol yang digunakan.

Extended ACL menggunakan penomoran dari 100 sampai 199. Berikut ini sintaks dari *Extended ACL*.

access-list access-list-number {deny | permit} protocol source-ip-address [wildcard-mask] destination-ip-address [wildcard-mask] operator

Dimana :

a. *access-list*
   Keyword yang digunakan untuk membuat ACL.
b. *access-list-number*
   Nomor yang menjadi identitas dari ACL, jangkauannya adalah 100-199.
c. *{permit | deny}*
   Tindakan terhadap paket data. Paket data dapat dilewatkan atau ditolak.
d. *protocol*
   Nama atau nomor protokol.
e. *source-ip-address*
   Alamat IP pengirim.
f. *destination-ip-address*
   Alamat IP tujuan.
g. *[wildcard-mask]*
   wildcard yang digunakan.
h. *operator*
   Operator yang digunakan.

Tabel 2.1 Operator untuk *Extended ACL*

Operator	**Penjelasan**
eq (*equal*)	Menentukan satu nomor *port*
neq (*not equal*)	Bentuk negasi atau kebalikan dari operator eq
gt (*greater than*)	Digunakan untuk menentukan jangkauan nomor *port* yang lebih besar dari nomor *port* yang diberikan
lt (*less than*)	Digunakan untuk menentukan jangkauan nomor *port* yang lebih kecil dari nomor *port* yang diberikan

Contoh :

access-list 100 permit tcp 202.101.51.3 0.0.0.0 host 172.16.1.1 0.0.0.0 eq 80

Extended ACL tersebut hanya akan meneruskan paket dari alamat IP 202.101.51.3 ke alamat IP 172.16.1.1 untuk protokol TCP dengan nomor *port* 80.

2.2.2. Memasang ACL pada *Interface*

Membuat aturan penyaringan paket data melalui baris-baris aturan pada ACL merupakan langkah awal dalam menggunakan ACL. Langkah selanjutnya, aturan penyaringan yang telah kita buat tersebut harus dipasang ke sebuah *interface* pada *router*.

Penyaringan terhadap paket data pada sebuah *interface* dapat dilaksanakan dalam dua arah, yaitu diterapkan untuk paket data yang masuk maupun yang keluar dari *router* melalui *interface* tersebut. ACL yang dipasang untuk menyaring paket data yang masuk melalui sebuah *interface* disebut ACL *inbound*. Sedangkan, ACL yang dipasang untuk menyaring paket data yang akan keluar melalui sebuah *interface* disebut ACL *outbound.*

ACL *inbound* akan menyaring paket data yang masuk melalui *interface router*. Saat sebuah paket data masuk, paket data tersebut akan diperiksa, yaitu dicocokkan dengan aturan-aturan yang ada pada ACL yang dipasang pada *interface* tempat paket data tersebut ingin masuk.

Jika paket data tersebut cocok dengan satu baris pernyataan *permit*, maka akan lansung diproses lebih lanjut, yaitu di-*routing* ke jaringan tujuan dari paket data. Jadi, paket data tersebut tidak akan dicocokkan lagi dengan baris-baris berikutnya dari ACL. Sebaliknya, jika paket data tersebut cocok dengan satu baris pernyataan *deny*, akan lansung ditolak, dan *router* akan mengirimkan paket ICMP *destination unreachable* kepada pengirim paket data. Jadi, paket data tersebut sudah tidak akan dicocokkan lagi dengan baris-baris berikutnya.

ACL *outbound* akan menyaring paket data yang akan keluar melalui *interface* Router. Saat sebuah paket data akan keluar, yakni minta di-*routing* keluar dari jaringan internal, paket data tersebut akan dicocokkan dengan aturan-aturan yang ada pada ACL yang dipasang pada *interface* tempat paket data tersebut ingin keluar.

Jika paket data tersebut cocok dengan satu baris pernyataan *permit*, ia akan lansung diproses lebih lanjut, yakni di-*routing* ke jaringan tujuan dari paket. Jadi, paket data tersebut tidak akan dicocokkan lagi dengan baris-baris berikutnya dari ACL. Sebaliknya, jika paket data tersebut cocok dengan satu baris pernyataan *deny*, maka akan langsung ditolak, dan *router* akan mengirimkan paket ICMP *destination unreachable* kepada pengirim paket. Jadi, paket data tersebut sudah tidak akan dicocokkan lagi dengan baris-baris berikutnya.

Pembuatan ACL dan pemasangannya pada *interface* memiliki beberapa aturan penting yang harus diperhatikan, antara lain:

1. Hanya satu ACL untuk satu protokol untuk satu arah penyaringan.
2. *Standard ACL* seharusnya dipasang sedekat mungkin dengan lokasi tujuan paket data.
3. *Extended ACL* seharusnya dipasang sedekat mungkin dengan lokasi asal paket data.
4. Sudut pandang arah penyaringan dilihat dari dalam Router, jadi *interface* Router dipandang sebagai sebuah pintu masuk.
5. Setiap pernyataan diproses secara berurutan, mulai dari pernyataan pada baris pertama sampai baris terakhir.
6. Pada akhir baris dari setiap ACL ada baris *implicit deny*, berupa pernyataan *deny all* yang berfungsi untuk menolak setiap paket yang tidak memenuhi satupun kriteria pernyataan *permit* pada baris-baris di atasnya.
7. Isi ACL seharusnya melakukan penyaringan paket data dari khusus ke umum, misalnya sebuah *host* harus ditentukan dahulu aturannya baru aturan untuk sekelompok *host*.
8. Proses pemeriksaan atau pencocokan sesuai dengan kriteria yang ditentukan dilaksanakan terlebih dahulu sebelum keputusan *permit* atau *deny* diterapkan.
9. Jangan memanipulasi sebuah ACL yang sedang aktif pada sebuah *interface.*
10. Menghapus ACL dari sebuah *interface* yang aktif harus dilakukan dengan hati-hati, sebaiknya *interface* tersebut dinonaktifkan terlebih dahulu.
11. Jika paket data yang datang ke sebuah *interface* ditolak, ACL akan mengirimkan pesan ICMP *Destination Unreachable* kepada pengirim paket tersebut.

2.3. Metode *Vulnerabilities Taxonomi*

Menurut Krsul (1998), "*taxonomi* merupakan kajian teoritis mengenai klasifikasi, termasuk landasan dasar, prinsip, prosedur, dan aturan-aturan yang berkaitan dengan klasifikasi

tersebut". Metode klasifikasi itu sendiri dapat digunakan untuk mempermudah analisa terhadap objek atau permasalahan yang akan diteliti.

Umumnya, orang mengenal istilah *taxonomi* dalam disiplin Ilmu Biologi, namun dalam disiplin Ilmu Komputer, khususnya bidang kajian Keamanan Sistem Informasi, wacana yang berkaitan dengan metode *taxonomi* telah dikenal sejak awal perkembangan ilmu ini. Menurut Wright (2007), "kebutuhan akan dibuatnya sebuah *taxonomi* yang terstruktur (sistem penamaan) untuk istilah-istilah atau layanan-layanan dalam dunia keamanan sistem informasi bukanlah hal baru. Semua layanan tersebut telah tersedia sejak dunia bisnis dan pemerintahan mulai menggunakan komputer, yakni sekitar tahun 70-an".

Salah satu contoh metode *taxonomi* dalam bidang kajian Keamanan Sistem Informasi adalah metode *taxonomi vulnerabilities*, yaitu metode sistematis yang digunakan untuk menjelaskan kelemahan-kelemahan sebuah sistem yang dapat dieksploitasi oleh orang-orang yang tidak berwenang terhadap sistem tersebut, serta metode penanggulangannya.

Menurut Krsul (1998), "Fungsi dari *taxonomi* adalah agar pemisahan atau pengurutan spesies dapat dilaksanakan sehingga dapat dibuat sebuah generalisasi yang telah mencakup seluruh spesies tersebut. Jadi, dapat kita katakan bahwa *taxonomi* memiliki nilai penjelasan. Taksonomi dapat juga digunakan untuk membuat prediksi adanya spesies-spesies lain yang belum dikenal dengan mempelajari pola dari spesies-spesies yang telah dikenal". Jadi, dapat kita katakan bahwa *taxonomi* memiliki nilai prediksi".

Dengan demikian, dapat disimpulkan bahwa metode *taxonomi vulnerabilities* sangat bermanfaat untuk menganalisa dan mengklasifikasikan kelemahan-kelemahan sebuah sistem, atau sebuah perangkat keamanan sistem, misalnya *firewall*. Dengan menggunakan metode *taxonomi vulnerabilitas*, kita tidak hanya dapat mendefinisikan penyebab-penyebab kelemahan, akibat buruk dari kelemahan tersebut, dan teknik penanggulangannya, tetapi kita juga dapat mengelompokkan beragam kelemahan tersebut sehingga memudahkan analisa yang tidak hanya dapat digunakan untuk menyelesaikan permasalahan yang ada saat ini, namun juga menjadi landasan dalam memecahkan permasalahan yang datang dikemudian hari.

Kamara, dkk (2003) mengklasifikasikan penyebab *vulnerabilities* yang berkaitan dengan *firewall* menjadi tujuh, yaitu:

1. Kesalahan Validasi *(Validation Error). Validation Error* terjadi saat program, perangkat, atau sistem berinteraksi dengan lingkungannya, yakni dalam rangka mengolah data-data yang datang dari lingkungannya, tanpa memeriksa terlebih dahulu keabsahan dari data-data tersebut.

   Ada tiga jenis data yang memerlukan validasi, yaitu *input*, *origin*, dan *target.* Validasi *Input*, artinya memeriksa bahwa data masukan tidak menyimpang, yakni benar-benar sesuai dengan yang diharapkan atau yang seharusnya, baik dalam hal nomor urutnya, jenis datanya, maupun formatnya. Validasi *origin*, artinya memeriksa bahwa data yang diolah benar-benar asli sesuai dengan apa yang dinyatakan oleh data tersebut. Validasi *target*, artinya memeriksa bahwa data hasil pengolahan diberikan kepada penerima yang berhak atas data tersebut. Tidak hanya itu, validasi *target* juga harus dapat meyakinkan bahwa data tersebut tidak diberikan kepada pihak yang tidak berhak.

2. Kesalahan Autorisasi *(Authorization Error). Authorization Error* disebut juga kesalahan otentikasi. Kesalahan ini terjadi saat pihak yang tidak berhak atau tidak berwenang diijinkan untuk melaksanakan operasi terhadap program, perangkat, atau pun sistem.

3. Kesalahan *Serialisasi/Aliasing (Serialization/Aliasing Error). Serialization Error* terjadi saat muncul eksploitasi terhadap sistem akibat perilaku *asinkron* dari dua sistem yang berbeda diijinkan untuk dioperasikan dalam waktu yang bersamaan. Sedangkan, *Aliasing Error* terjadi saat ada dua nama untuk sebuah objek yang sama dapat mengakibatkan perubahan pada isi objek tersebut secara tidak terduga, sehingga konsekuensinya adalah dapat mengakibatkan perubahan validasi yang sebelumnya telah diaplikasikan kepada objek tersebut.

4. Kesalahan Pengecekan Batasan Sistem *(Boundary Checking Error). Boundary Checking Error* muncul akibat kegagalan dalam memeriksa batasan-batasan yang diperbolehkan, yakni pelanggaran terhadap batasan-batasan yang telah ditetapkan. Akibatnya, terjadi *Buffer Overflow.*

5. Kesalahan *Domain (Domain Error). Domain*

Error terjadi saat muncul celah keamanan pada sebuah *Domain*, yakni batasan *Domain* tersebut dilanggar, sehingga mengakibatkan adanya informasi yang seharusnya hanya boleh diakses oleh pengguna pada *Domain* tersebut bocor ke pihak luar yang tidak berhak.

6. Rancangan yang lemah atau kesalahan Rancangan *(Weak/Design Error). Weak/Design Error* terjadi saat tahapan proses perancangan sistem. Contoh dari kesalahan rancangan misalnya lemahnya algoritma enkripsi dimana hasil enkripsinya, yakni *cipher text*-nya mudah dipecahkan atau dilaksanakan *kriptoanalisis* terhadapnya.
7. Kesalahan-kesalahan lainnya. Kesalahan-kesalahan lain yang tidak masuk ke dalam enam kategori kesalahan sebelumnya masuk ke dalam kategori ini.

2.4. Keterbatasan ACL

Cisco System Inc. membuat ACL sebagai perangkat pengamanan dasar pada jajaran produk Router mereka. ACL dibuat untuk memberikan fungsi penyaringan paket data yang merupakan fungsi mendasar dari sebuah *firewall.* ACL dirancang untuk memberikan fungsi pengamanan yang optimal, namun tetap sederhana dalam pembuatan dan pengimplementasiannya.

Sebuah sistem *firewall* yang kokoh karena dibangun dari beragam fungsi pengamanan sekalipun, tetap bukan merupakan solusi tunggal dalam mengamankan sebuah sistem. Apalagi ACL yang hanya melaksanakan penyaringan paket data saja dalam melaksanakan fungsi pengamanan. Sebelum menggunakan ACL, penting untuk diketahui keterbatasan-keterbatasan dari ACL, sehingga dapat dilaksanakan tindakan yang diperlukan agar keamanan sistem tetap terjaga secara optimal. Berikut ini keterbatasan-keterbatasan dari ACL dan analisa berikut solusinya sekaligus klasifikasinya berdasarkan metode *taxonomi vulnerabilitas*.

1. Pembuatan ACL harus dilaksanakan secara berurutan atau *sekuensial.* Saat pernyataan-pernyataan pada ACL jumlahnya makin banyak, hal ini tidak hanya akan sangat merepotkan dalam pembuatannya, namun juga sangat sulit untuk melaksanakan audit dan merawatnya.

Kesalahan yang mungkin muncul pada ACL yang memiliki banyak baris aturan adalah permasalahan logik, yakni sulit untuk tetap menjaga konsistensi logik dari seluruh baris ACL tersebut. Ada beberapa kesalahan logik yang dapat muncul, misalnya baris aturan yang terduplikasi *(redundant),* baris aturan yang saling bersilangan *(intersection),* dan baris aturan yang tidak konsisten.

Baris aturan yang terduplikasi *(redundant),* artinya ada dua atau lebih baris aturan yang memiliki makna atau aturan yang sama, yakni sebuah baris aturan ternyata merupakan bagian atau *subset* dari baris aturan lain yang lebih lengkap.

Baris aturan yang saling bersilangan *(intersection),* artinya ada dua atau lebih baris aturan yang memiliki makna atau aturan yang saling bersilangan, yakni baris aturan yang satu memiliki aturan yang telah diatur oleh baris aturan lain atau aturan penyaringan mereka saling beririsan.

Baris aturan yang tidak konsisten, artinya ada dua atau lebih baris aturan yang memiliki makna atau aturan yang saling berlawanan atau bertolak belakang, yakni adanya sebuah baris aturan yang menyatakan *permit* untuk sebuah paket data sementara ada baris aturan lain yang menyatakan *deny* untuk paket data yang sama atau sebaliknya.

Kesalahan-kesalahan logik tersebut muncul umumnya karena makin berkembangnya ACL, yakni makin bertambahnya baris aturan pada ACL tersebut yang disebabkan oleh makin berkembangnya jaringan dan adanya perubahan atau penyesuaian aturan kebijakan keamanan mengakibatkan makin sulitnya menyesuaikan baris-baris aturan yang telah ada dengan aturan-aturan baru yang harus ditambahkan. Diperlukan pemahaman yang baik mengenai aturan pembuatan ACL dan ketelitian yang tinggi serta ketekunan untuk membuat ACL yang akurat dan efisien.

Sebenarnya, Cisco telah menyediakan alat bantu yang dapat digunakan untuk manajemen ACL, yaitu Cisco *Works*. Cisco *Works* merupakan aplikasi berbasis *Graphical User Interface* (GUI) yang dapat digunakan sebagai alat bantu untuk melakukan pemantauan dan mengatur perangkat-perangkat jaringan milik Cisco.

Untuk manajemen ACL, Cisco *Works* menyediakan fasilitas ACL *Manager* yang dapat digunakan untuk membuat *(create)*, mengubah *(edit)*, dan mengatur urutan baris-baris aturan pada ACL serta fungsi-fungsi yang berkaitan dengan manajemen ACL lainnya.

Berdasarkan *taxonomi vulnerabilitas*, keterbatasan ACL ini masuk ke dalam kesalahan atau kelemahan desain dimana akibat dari keterbatasan ini adalah menyulitkan para penggunanya. Meskipun demikian, secara objektif, dimana Cisco merancang ACL agar sederhana untuk menjamin kinerja yang tinggi maka administrator jaringan yang diharapkan dapat memahami dengan baik ACL agar efek dari keterbatasan ACL ini dapat diminimalkan dan keuntungan dari kesederhanaan dan unjuk kerja yang tinggi dari ACL didapatkan.

2. ACL merupakan *packet filtering firewall* yang bersifat *stateless*, artinya ACL tidak dapat memeriksa hakikat dari setiap koneksi yang sedang terjalin. Selama paket-paket data yang meminta lewat memenuhi persyaratan yang tercantum pada ACL, paket data tersebut akan dilewatkan. Jadi, bisa saja sebuah paket data yang memiliki alamat IP yang sah yang diperbolehkan lewat oleh ACL ternyata merupakan paket data dari penyerang yang sedang berusaha untuk mengeksploitasi sistem, yakni si penyerang tadi telah melakukan *IP Spoofing*.

   Berdasarkan *taxonomi vulnerabilitas*, keterbatasan ACL ini masuk ke dalam *validation error* yang berkaitan dengan validasi *origin* dimana ACL tidak mampu atau salah dalam memeriksa kebenaran alamat IP dari paket data yang lewat akibat *IP Spoofing* yang dilakukan oleh penyerang, sehingga penyerang dapat masuk untuk mengeksploitasi jaringan dengan menyamarkan alamat IP yang dipergunakannya.

3. ACL hanya akan melaksanakan penyaringan terhadap paket-paket data yang ingin melewati sebuah *interface* dari *router*, yaitu paket data yang ingin masuk dari jaringan eksternal ke jaringan internal maupun sebaliknya. ACL tidak dapat menyaring paket data yang berasal dari *router* itu sendiri. Misalnya, paket-paket data *routing protocol*, seperti *OSPF hello*, paket *routing update* dan sejenisnya yang digunakan oleh *router* untuk saling berbagi informasi *routing* akan dilewatkan begitu saja oleh ACL tanpa perlu diperiksa.

   Penyerang dapat membuat paket *route advertisement* yang berisi rute-rute yang dapat melewatkan paket-paket data ke jaringan miliknya. Jika *router* yang kita gunakan menggunakan protokol *routing* yang dinamis, dan umumnya memang protokol *routing* dinamis yang digunakan karena kemudahan administrasinya dibandingkan harus mengisi tabel *routing* secara manual *(static routing)*, maka ACL tidak mampu menyaring mana paket *route advertisement* yang sah dengan yang tidak.

   Static routing memang lebih aman dibandingkan *Dynamic Routing*. Menurut Brenton (2003), "Meskipun *routing statis* memerlukan *maintenance* atau pemeliharaan yang cukup banyak, namun *routing statis* adalah cara yang paling aman untuk membangun *routing table* Anda. *Routing* dinamis memungkinkan *routing table* untuk diperbaharui *(update)* secara dinamis oleh alat-alat di *network*. Seorang penyerang bisa mengeksploitasi fasilitas ini untuk memberikan informasi *routing* yang tidak benar kepada *router-router* kita yang bisa menghalangi *network* kita untuk bekerja dengan baik".

   Solusi dari masalah ini adalah penggunaan protokol *routing* yang memiliki fitur otentikasi dan enkripsi, seperti *Open Shortest Path First* (OSPF). Protokol OSPF mensyaratkan *router-router* yang berpartisipasi dalam pertukaran *routing table* untuk memberikan kata sandi agar informasi rute mereka dapat diterima. Informasi kata sandi tersebut berikut informasi *routing table* yang ingin diberikan dan kunci kriptografi yang digunakan dienkripsi dan disertakan dalam paket *update routing table*.

   Jadi, pengamanan pertukaran informasi rute diserahkan pada protokol *routing*. Tentu saja, protokol *routing* yang digunakan harus telah memiliki fitur keamanan seperti OSPF. Tanpa adanya fasilitas keamanan, *router* yang kita miliki rentan terhadap masuknya informasi-informasi rute yang ilegal. Namun, seorang pengelola jaringan komputer tentu saja selalu dapat dengan

mudah memeriksa secara manual *routing table* dari *router-router* yang berada dalam pengelolaannya menggunakan perintah-perintah yang disediakan IOS melalui *console* menggunakan sebuah *terminal*.

Berdasarkan *taxonomi vulnerabilitas*, keterbatasan ACL ini masuk ke dalam kesalahan atau kelemahan desain dimana akibat dari keterbatasan ini menyebabkan rentannya Router terhadap masuknya informasi-informasi rute yang tidak benar hasil dari paket *routing update* dari pihak-pihak yang tidak terpercaya.

4. Saat sebuah paket telah memenuhi persyaratan dari sebuah baris pernyataan pada ACL, baik pernyataan *permit* maupun *deny*, tindakan yang sesuai akan segera dilaksanakan. Jadi, paket tersebut tidak akan dicocokkan lagi dengan baris-baris berikutnya. Artinya, ACL harus dibuat dengan sangat teliti agar jangan sampai salah dalam mengambil tindakan terhadap sebuah paket data yang dapat mengakibatkan paket data yang seharusnya ditolak menjadi diteruskan dan sebaliknya.

   Meskipun tidak mutlak sebagai *design error* berdasarkan *taxonomi vulnerabilitas*, hal ini merupakan keterbatasan rancangan dari ACL, dimana ACL dirancang agar sederhana dan cepat sehingga operasinya tidak terlalu mengganggu kinerja dari *router*. Jadi, begitu sebuah paket telah cocok dengan satu baris aturan dari ACL, baris-baris berikutnya tidak perlu diperiksa lagi agar tidak menghabiskan sumber daya pemrosesan dari *router*.

5. ACL tidak dirancang untuk mendeteksi penyerang dari dalam, yaitu pengguna yang sah dari jaringan internal yang memanfaatkan sumber daya jaringan untuk melaksanakan tindakan jahatnya. Jadi, seorang pengguna jaringan yang sah dapat melakukan kegiatan apapun yang diinginkannya dalam jaringan tanpa terdeteksi oleh ACL.

   Untuk dapat menganalisa paket-paket data yang hilir mudik dalam jaringan internal diperlukan sebuah program *network analyzer* atau *traffic analyzer.* Program *analyzer* tersebut dapat menangkap setiap paket data yang hilir mudik dalam jaringan dan menganalisanya, yaitu dengan membandingkannya dengan *database* modus operandi serangan yang dimilikinya sehingga dapat ditentukan aktivitas apa yang sedang dilaksanakan oleh pemilik paket data tersebut.

   Jika ternyata terbukti bahwa paket data tersebut mencurigakan, yaitu cocok dengan modus operandi yang terdaftar dalam *database*, maka dapat segera diambil tindakan yang semestinya, misalnya pemutusan koneksi secara lansung atau dikirimkannya laporan kepada pengelola jaringan agar pengelola jaringan tersebut dapat segera melaksanakan tindakan yang diperlukan. Inilah yang dilaksanakan oleh perangkat keamanan yang dikenal dengan nama *Intrusion Detection System* (IDS).

   IDS tidak lagi memperdulikan apakah paket data tersebut menggunakan alamat IP yang sah atau tidak. Selama aktivitas koneksi yang sedang dibangun oleh paket data tersebut cocok dengan modus operandi atau pola serangan yang terdaftar dalam *database* IDS, koneksi yang dibangun oleh paket data tersebut adalah koneksi yang ilegal dan setiap paket data yang berkaitan dengannya harus ditahan.

   Berdasarkan *taxonomi vulnerabilitas*, keterbatasan ACL ini masuk ke dalam kesalahan atau kelemahan desain, yaitu ACL adalah *packet filtering* yang bersifat *stateless.*

6. ACL tidak dapat mengenali *malware (malicious ware)*, seperti : *virus*, *worm*, *trojan horse*, dan sejenisnya. *Malware* merupakan alat yang sangat ampuh yang sering menjadi senjata pamungkas oleh penyerang untuk mengeksploitasi bahkan melumpuhkan sebuah sistem. ACL dapat saja melewatkan sebuah paket yang merupakan sebuah *malware*, selama paket tersebut memenuhi persyaratan *permit* pada ACL.

   ACL hanya mampu menerapkan *access control* melalui mekanisme *filtering*. ACL tidak mampu mengenali apalagi menangani *malware.* Menurut Brenton (2003), "*Access Control* tidak akan menghilangkan atau mendeteksi keberadaan sebuah program kosmetik. *Access control* hanya sebuah metode untuk membantu sistem kita menahan infeksi *virus"*. Maksudnya, metode *access control* yang kita terapkan, baik melalui mekanisme *filtering*, otentikasi, dan sejenisnya hanya akan menghalangi

malware untuk masuk. Ia tidak akan mampu mendeteksi *malware* apalagi memperbaiki sumber daya yang telah terinfeksi oleh *malware*. Kita membutuhkan aplikasi lain untuk melakukannya, yaitu aplikasi *antimalware*.

ACL hanya dapat menghalangi *malware* untuk masuk dengan berasumsi bahwa *malware* hanya dibawa oleh paket data yang tidak diijinkan untuk lewat, yaitu : asumsi bahwa aturan pada ACL memang telah dirancang untuk hanya melewatkan paket data yang bersih dan terpercaya. Jadi, jika seorang penyerang ingin menyebarkan *malware* maka ia dapat melakukannya dengan menyamarkan paket data yang berisi m*alware* sehingga seolah-olah merupakan paket data yang bersih dan terpercaya, misalnya paket data tersebut memiliki alamat IP yang terpercaya, yaitu penyerang melakukan *IP Spoofing*.

Berdasarkan *taxonomi vulnerabilitas*, secara prinsip keterbatasan ACL ini masuk ke dalam kesalahan lain-lain. Namun, berkaitan dengan *IP Spoofing* sebagai modus operandinya, keterbatasan ini dapat masuk ke dalam kesalahan validasi.

7. *Standard* dan *Extended* ACL tidak memiliki metode otentikasi. Metode otentikasi yang diterapkan bagi pengguna digunakan untuk menguji keabsahan seorang pengguna yang ingin mengakses sumber daya jaringan.

   Umumnya, metode otentikasi diterapkan menggunakan pasangan *user name* dan *password*. Jadi, pengguna yang ingin mengakses sumber daya sistem terlebih dahulu harus memberikan *user name* dan *password* yang dimilikinya. Mekanisme otentikasi yang diterapkan pada sistem tersebut kemudian melaksanakan *query* ke database yang sesuai untuk mencari *account* yang cocok. Jika ditemukan, pengguna tersebut baru boleh mengakses sumber daya sistem, sebaliknya jika tidak ada yang cocok, maka pengguna tersebut tidak diijinkan untuk mengaksesnya.

   Karena *Standard* dan *Extended* ACL tidak memiliki metode otentikasi, maka tidak ada mekanisme pengujian terhadap pengguna yang ingin mengakses sumber daya jaringan. Siapapun boleh mengakses sumber daya jaringan selama ybs memenuhi persyaratan pada aturan yang terdapat pada ACL, misalnya alamat IP yang sah untuk digunakan pada jaringan tersebut. Dengan demikian, seorang penyerang yang telah berhasil mendapatkan akses ke jaringan sudah tidak ada bedanya lagi dengan pengguna jaringan yang sah, karena pada dasarnya pengguna jaringan yang sah juga tidak memiliki *user account*, yakni pada jaringan tersebut memang tidak dikenal adanya *user account* untuk membedakan antara pengguna jaringan yang sah dengan yang tidak. Kesimpulannya, tanpa adanya metode otentikasi, maka tidak ada mekanisme untuk membedakan antara pengguna jaringan yang sah dengan pengguna ilegal, konsekuensinya adalah tidak dapat diterapkannya pembatasan hak akses atau *access control* berdasarkan *user account*.

   Berdasarkan *taxonomi vulnerabilitas*, keterbatasan ACL ini masuk ke dalam *authorization error* dimana ACL tidak mampu untuk melaksanakan otentikasi terhadap permintaan koneksi sehingga tidak dapat dibedakan antara pengguna yang sah dan pengguna ilegal.

III. PENUTUP

3.1. Kesimpulan

Berdasarkan analisa yang telah dijabarkan pada bagian sebelumnya, dapat diambil beberapa kesimpulan, antara lain:

1. Salah satu teknik untuk mengamankan sistem adalah dengan konsep *Access Control* atau pengendalian akses.
2. Salah satu contoh penerapan *Access Control* pada Router adalah Cisco IP *Access Control List* (ACL) yang diterapkan oleh Cisco Systems Inc. pada jajaran produk Router mereka.
3. *ACL* adalah sebuah daftar berurut yang paling sedikit terdiri dari satu pernyataan *permit* dan mungkin satu atau lebih pernyataan *deny*.
4. Sebuah paket data akan diteruskan jika memenuhi kriteria pada pernyataan *permit* dan tidak memenuhi kriteria pada pernyataan *deny* dan sebaliknya paket data tidak akan diteruskan jika tidak memenuhi kriteria pada pernyataan *permit* atau memenuhi kriteria pernyataan *deny*.
5. Daftar pernyataan pada *ACL* diproses secara

terurut atau *sekuensial* dari baris pertama hingga baris terakhir, dimana jika sebuah paket data telah cocok dengan sebuah pernyataan, baik *permit* ataupun *deny*, tindakan yang sesuai akan segera dilaksanakan tanpa perlu membandingkan paket data tersebut dengan baris-baris berikutnya.

6. Ada dua jenis *IP Access Control List*, yaitu *Standard ACL* dan *Extended ACL*.
7. *Standard ACL* hanya menyaring paket data berdasarkan alamat IP pengirim paket.
8. *Extended ACL* dapat menyaring paket data berdasarkan alamat IP pengirim, alamat IP tujuan, nomor *port*, dan jenis protokol yang digunakan.
9. ACL yang telah dibuat harus dipasang pada sebuah *interface* dari Router sekaligus ditentukan arah penyaringan yang dilaksanakan oleh ACL tersebut.
10. ACL yang dipasang untuk menyaring paket yang masuk ke sebuah *interface* disebut ACL *inbound*. Sedangkan, ACL yang dipasang untuk menyaring paket yang akan keluar melalui sebuah *interface* disebut ACL *outbound*.
11. ACL dapat dipasang di setiap *interface* dari Router dengan syarat satu ACL untuk satu protokol dan satu arah penyaringan.
12. *Standard ACL* seharusnya dipasang sedekat mungkin dengan tujuan paket data untuk mencegah pemblokiran paket data yang sah.
13. *Extended ACL* seharusnya dipasang sedekat mungkin dengan asal paket data untuk menghemat *bandwith* jaringan, yakni paket data tersebut tidak perlu masuk ke dalam jaringan jika memang harus ditolak.
14. Pada akhir baris dari setiap ACL ada baris *implicit deny*, berupa pernyataan *deny all* yang berfungsi untuk menolak setiap paket yang tidak memenuhi satupun kriteria pernyataan *permit* pada baris-baris di atasnya.
15. Jika sebuah ACL tidak memiliki satupun pernyataan, baik pernyataan *permit* maupun *deny*, maka ACL tersebut tidak akan melewatkan paket data apapun.
16. Metode *taxonomi* vulnerabilitas dapat digunakan untuk menganalisa dan mengklasifikasikan kelemahan-kelemahan sebuah sistem secara sistematis, sehingga tidak hanya dapat digunakan untuk menyelesaikan permasalahan yang ada saat ini, namun juga menjadi landasan dalam memecahkan permasalahan yang datang di kemudian hari.
17. *ACL* dapat digunakan untuk membuat *Packet Filtering Firewall* yang akan melaksanakan penyaringan terhadap setiap paket data.
18. ACL bukan merupakan solusi total untuk membangun sebuah sistem *Firewall*, apalagi sebagai pengaman tunggal sebuah sistem.
19. Makin bertambahnya baris pernyataan pada sebuah ACL, makin sulit menjaga konsistensi aturan ACL tersebut sehingga kesalahan makin mungkin terjadi.
20. Proses penyaringan paket data oleh ACL memerlukan sumber daya pemrosesan milik Router sehingga akan menurunkan kinerja Router secara keseluruhan.
21. ACL bersifat *stateless*, sehingga tidak dapat memeriksa status dari setiap koneksi yang sedang terjalin pada jaringan.
22. ACL tidak dapat mengenali apalagi menanggulangi serangan *malware*.
23. ACL sangat rentan terhadap *IP Spoofing* atau pemalsuan alamat IP.
24. ACL tidak memiliki metode otentikasi sehingga tidak dapat membedakan antara pengguna yang sah dengan yang tidak.

3.2. Saran

Berikut ini saran-saran yang dapat dikemukakan berkaitan dengan penggunaan ACL sebagai perangkat pengamanan.

1. Jika aturan penyaringan paket data masih sederhana, gunakanlah *Standard ACL* agar pembuatan, perawatan, dan auditnya lebih mudah.
2. Saat aturan penyaringan paket data makin rumit, gunakanlah *Extended ACL* agar aturan tersebut dapat terakomodasi dengan baik.
3. Sebelum membuat ACL, rancang terlebih dahulu aturan penyaringan paket data yang diinginkan dengan teliti.
4. Jangan menerapkan ACL pada Router kecuali Anda telah yakin bahwa baris-baris aturan pada ACL tersebut telah sesuai dengan aturan yang ingin Anda terapkan.
5. Untuk menutupi kekurangan ACL dalam hal ketidakmampuannya untuk mendeteksi paket-paket data yang berbahaya yang berhasil menyusup masuk ke jaringan internal, gunakanlah IDS (*Intrusion Detection System*).
6. Gunakanlah NAT *(Network Address Translation)* dan PAT *(port Address Translation)* yang juga merupakan fitur

keamanan yang terdapat pada Router Cisco untuk mendukung fungsi pengamanan dari ACL, yakni dengan menyembunyikan alamat-alamat IP yang digunakan pada jaringan internal.

7. Untuk menutupi kekurangan ACL dalam hal ketidakmampuannya untuk melaksanakan otentikasi, Anda dapat menggunakan RADIUS *(Remote Authentication Dial-In User Service)* atau TACACS+ *(Terminal Access Controller Access Control System)* dimana dukungan terhadap keduanya telah disediakan dengan baik oleh Cisco.
8. Gunakanlah juga perangkat keamanan lain untuk mendukung ACL, seperti *Proxy Server,* aplikasi *antimalware, bastion host,* dan sebagainya.
9. Untuk pengamanan yang lebih optimal dan jika tersedia sumber daya manusia yang cakap gunakanlah *honeypot* untuk mengalihkan setiap usaha serangan yang masuk.
10. Lakukanlah *update* terhadap Cisco IOS dan kunjungi situs resmi mereka untuk mendapatkan *white paper* dan berbagai laporan penting lainnya yang berkaitan dengan peralatan yang Anda gunakan.
11. Kunjungilah situs-situs yang memberikan informasi aktual berkaitan dengan keamanan jaringan komputer untuk mengantisipasi jenis serangan, dan celah keamanan serta *bug* yang berkaitan dengan perangkat-perangkat yang Anda gunakan.
12. Gunakanlah aplikasi pembantu untuk manajemen ACL, seperti Cisco *Works* agar manajemen ACL lebih mudah.

DAFTAR PUSTAKA

Al-Wabel Abdulelah A. dan Al-Shayea Hamid I. 2009. ACL *Analysis Tool*. King Saud *University*. Arab Saudi.

Benardi, Beny. 2004. Membangun *Firewall* dengan Cisco *Router*. Penerbit PT Elex Media Komputindo. Jakarta.

Brenton, Chris dan Hunt, Cameron. 2005. *Network Security*. Penerbit PT Elex Media Komputindo. Jakarta.

Cisco *System*, Inc. Cisco IOS *Security Configuration Guide Release 12.2SX* [http://www.cisco.com/en/US/docs/ios/security/configuration/guide/12_2sx/sec_12_2sx_book.html] (Accessed November 17, 2008)

Cisco System, Inc. TACACS+ and RADIUS Comparison [http://www.cisco.com/en/US/tech/tk59/technologies_tech_note09186a0080094e99.shtml] (Accessed January 14, 2008)

Habtamu, Abie. 2000. *An Overview of Firewall Technologies*. Norwegian Computing Center. Norwegia.

Ivan Victor, Krsul. 1998. *Software Vulnerability Analysis*. Purdue University. Amerika Serikat.

Kamara, Seny dkk. 2003. *Analysis of Vulnerabilities in Internet Firewalls*. Purdue University. Amerika Serikat.

Wright, Craig S.. 2007. A Taxonomy of Information Systems Audits, Assessments and Reviews. SANS Institute. Amerika Serikat

INSTALLASI DESAIN DAN PENGEMBANGAN JARINGAN KOMPUTER BERSKALA KECIL
(STUDI KASUS : INSTALASI *ONLINE WIRELESS* DI RUMAH)

Agus Dendi Rachmatsyah

AMIK Bina Sarana Informatika
Jl. Salemba Raya No. 45 Jakarta Pusat, Indonesia
agus_dnd@yahoo.com

Abstract
Computer and network technology evolve very rapidly and is advancing. There are a lot of connected computers in the network which facilitate effective and efficient working activities. This also increases the efficiency in computer usage since many resources are available at the same time. In this writing, the author uses library research and direct observation. Based on the case study done on the installation design of small-scale computer network at home and at work, it can be concluded that the small-scale computer network is very appropriate to be applied in housing and in the community. It is due to the increasing needs on technology in Indonesia, especially in the field of computers and wireless technology, simple computers without a LAN card and the wireless (wifi) suitable for workplaces and homes.

Keyword: wireless, network, internet protocol

I. PENDAHULUAN

Pada zaman modern seperti saat ini, komputer merupakan suatu alat yang tidak asing lagi, yang tidak ubahnya seperti *ponsel*, motor, mobil dan sebagainya. Hal ini merupakan suatu perubahan dan kemajuan dari teknologi informasi. Komputer telah berubah dari piranti yang hanya khusus untuk dunia bisnis hingga *piranti* serba guna yang dapat digunakan untuk multi kegiatan, seperti melakukan komunikasi *realtime, streaming video, audio,* dan dapat terkoneksi ke internet yang memungkinkan komputer mengakses informasi informasi dari seluruh dunia. Komputer tidak lagi dianggap sebagai alat untuk menghitung saja, namun juga telah menjadi komponen *integral* untuk bekomunikasi, menikmati hiburan dan alat bantu untuk pendidikan. Jaringan komputer dapat di kelompokkan menjadi tiga kelompok yaitu *Local Area Network (LAN), Metropolitan Area Network (MAN)* dan *Wide Area Network (WAN).* Menurut Ali Zaki dan SmitDev Community (2008) berdasarkan topologinya ada tiga macam jaringan:

1. *Bus Network*
   Ciri topologi *bus* adalah adanya *backbone* atau batang utama yang terkait ke komputer-komputer yang terhubung dengan jarak tertentu. Topologi *bus* dianggap sebagai topologi yang pasif karena komputer yang tergabung ke *bus* hanya diam dan mendengarkan (melalui *network interface card*). Ketika ada data, maka akan menerima data tersebut. Jika komputer ingin mengirimkan data ke komputer lain di jaringan, komputer akan menunggu terlebih dahulu sampai tidak ada orang yang mengirimkan informasi di *bus*. Jaringan *bus* umumnya memakai kabel jaingan *koaksia*, bentuknya hamper mirip dengan kabel *koaksial* untuk televisi namun ada sedikit perbedaan. Kabel-kabel tersebut dihubungkan ke komputer menggunakan *konektro T*, dan tiap ujung dari *bus* jaringan ditutup menggunakan *terminator* yang sesuai dengan jenis kabel yang digunakan. Kelemahan dari topologi bus adalah ukuran jaringan terbatas karena dibatasi oleh jangkauan kabel untuk memindahkan data. Kelebuhan dari jaringan *bus* adalah kemudahan dalam proses pembuatan dan tidak memerlukan biaya yang banyak untuk ukuran jaringan yang sedikit.
2. *Star Network*
   Di jaringan *star*, komputer-komputer di jaringan saling terhubung karena adanya *piranti sentral* yang bernama *hub*. Tiap komputer terhubung ke *port-port* di *hub* dengan kabel (umumnya kabel yang digunakan adalah *UTP*). Karena topologi *star* menggunakan kabel terpisah untuk setiap komputer maka jaringan *star* mudah untuk diperluas. Batasan yang ada adalah jumlah *port* yang dapat diakomodasi oleh *hub*

tersebut. Untuk menambah jumlah jaringan baru di jaringan *star* juga sangat mudah karena hanya menambahkan kabel baru antara komputer dan hub. Kerugian menggunakan topologi jaringan *star* adalah mengenai perkabelan dan *hub*. Keuntungan terbesar penggunaan topologi *star* adalah mudahnya menambah komputer-komputer baru ke dalam jaringan.

3. *Ring Network*
   Sebuah topologi *ring* menghubungkan komputer-komputer di *LAN* menggunakan kabel secara melingkar. Topologi *ring* menggerakkan informasi di kabel dalam satu arah. Komputer di jaringan mengirim ulang paket-paket data ke komputer berikutnya di *ring*. *Hardware token ring* saat ini sangat mahal dan memerlukan keahlian yang mendetail mengenai *token ring*.

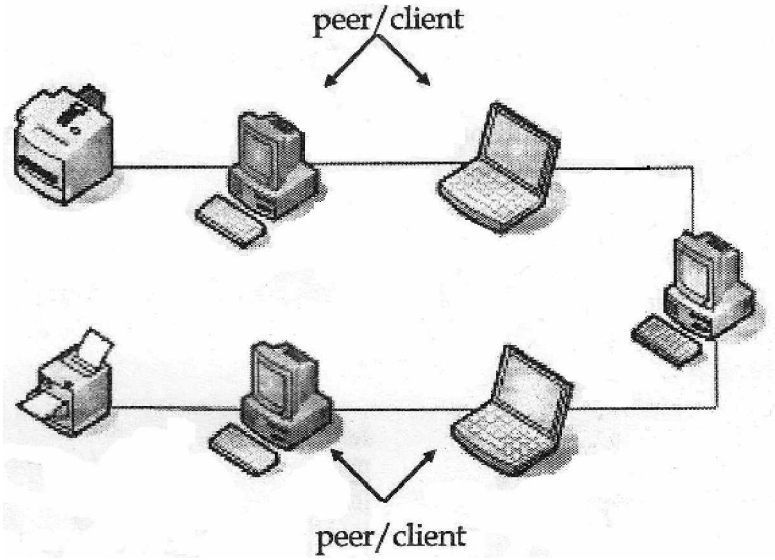
METODE PENELITIAN

Dalam pembuatan tulisan ini penulis menerapkan beberapa metode pengumpulan data antara lain:
1. Studi Pustaka dimana pada Metode ini penulis, mengutip beberapa definisi dari literatur-literatur yang berasal dari buku teks dan *browsing* internet.
2. Penulis menggunakan metode pengamatan langsung, eskperimental dimana metode ini penulis mencoba mempraktekan, menganalisa, mengadakan implementasi cara kerja dari sistem instalasi ini agar dapat di gunakan.

II. PEMBAHASAN

KONFIGURASI JARINGAN *PEER-TO-PEER*

Peer To Peer merupakan suatu kumpulan yang saling terhubung dalam jaringan dan beroperasi dengan bobot yang sama. Pada setiap komputer disebut dengan *peer*. Masing-masing *peer* berbagi *(Share)*, sehingga setiap komputer dapat beraksi sebagai *client* atau sebagai *server* atau keduanya. Jaringan *Peer To Peer* atau *P2P* adalah jaringan yang paling sederhana, tetapi perlu mengusai beberapa pengetahuan mendasar tentang bagaimana mengkonfigurasikan *protocol* jaringan dan melakukan *sharing printer* dan *file*.



Gambar 2.1: Model *Peer To Peer*

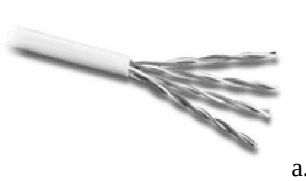
Menurut tulisan yang diambil dari *website.http://www.malangkab.go.id/kabmalang/galeri – ti Peer To Peer network* adalah jaringan komputer yang terdiri dari beberapa komputer (biasanya tidak lebih dari 10 komputer dengan 1-2 *printer*). Dalam sistem jaringan ini yang diutamakan adalah penggunaan program, data dan *printer* secara bersama-sama.

Jaringan *Peer To Peer* hanya disarankan ketika komputer yang ada di jaringan sedikit dan kecenderungan jumlah tersebut tidak akan bertambah besar di masa mendatang.

Menurut tulisan yang diambil dari website

http://www.situsinformasiinternet.com/2009/07/membuat-jaringan-peer-to-peer-pc-to-pc.html *untuk membuat jaringan komputer* peer to peer *kabel UTP yang dibuat harus dengan* Crossover / Crossline *karena jika menggunakan* Straight Through *kabel LAN dianggap tidak terkoneksi (*a network cable is unplugged*) kecuali jika* Ethernet *atau* LAN Card *yang anda gunakan sudah* support *dengan* straight through. *Untuk membuat kabel jaringan* Crossover/ Crossline *sebagai berikut :*

Siapkan alat-alat yang dibutuhkan :



a. Kabel UTP

b. *Konektor RJ-45*

c. *Crimping Tool*

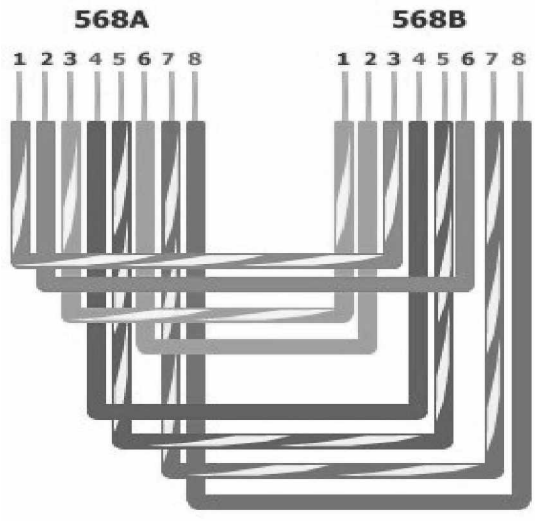
d. *LAN Tester*

Gambar 2.2. Alat-alat yang dibutuhkan

Perlu diketahui bahwa kabel UTP memiliki 4 pasang kabel kecil di dalamnya yang memiliki warna berbeda, 4 pasang kabel itu adalah :

Pasangan 1 : Putih/Biru dan Biru,
Pasangan 2 : Putih/Oranye dan Orange,
Pasangan 3 : Putih/Hijau dan Hijau,
Pasangan 4 : Putih/Coklat dan Coklat

Proses pembuatan :



Gambar 2.3. Susunan Kabel

Urutan pemasangan : Salah satu sisi kabel dibuat sesuai dengan standar *"Straight Through"*, sedangkan sisi kabel lainnya, dilakukan *"Cross-Over"*, yaitu :

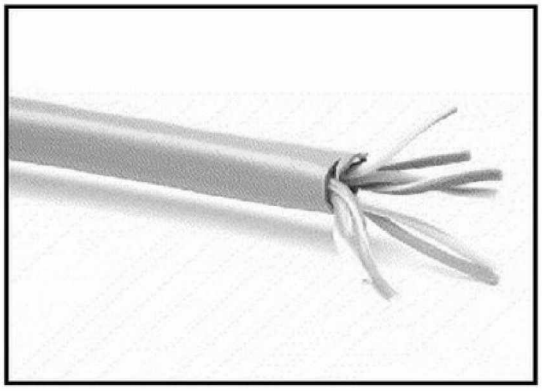
Pin 1 : Putih/Hijau
Pin 2 : Hijau
Pin 3 : Putih/Oranye
Pin 4 : Biru
Pin 5 : Putih/Biru
Pin 6 : Oranye
Pin 7 : Putih/Coklat
Pin 8 : Coklat

Harap diingat bahwa yang dibuat *crossover* hanya salah satu sisi kabel saja.

Langkah-langkah pemasangan kabel UTP pada konektor RJ45 :

1. Kupas jaket dari kabel UTP dengan menggunakan *crimping tool* atau alat pengupas kabel khusus.

2. Pisahkan empat lilitan kabel UTP menjadi delapan bagian, setelah itu luruskan tiap-tiap kabel agar dapat mudah dipotong.



Gambar 2.4. Lilitan kabel UTP

3. Susunlah urutan warna sesuai dengan konfigurasi *crossover* dan sesuaikan ujung kabel yang akan dipotong dengan konektor yang akan dipasang.

4. Gunakan tang pemotong atau *crimping tools*, potonglah ujung kabel secara rata agar kabel mudah dimasukan ke lubang konektor.

5. Masukkan ujung kabel yang telah dipotong ke lubang konektor RJ-45 secara bersamaan, kemudian jepit konektor dengan menggunakan *crimping tool* agar konektor terkunci.

6. Lakukan tes dengan *LAN Tester*, jika semua lampu indikator menyala berarti semua bagian kabel sudah terpasang dengan benar.

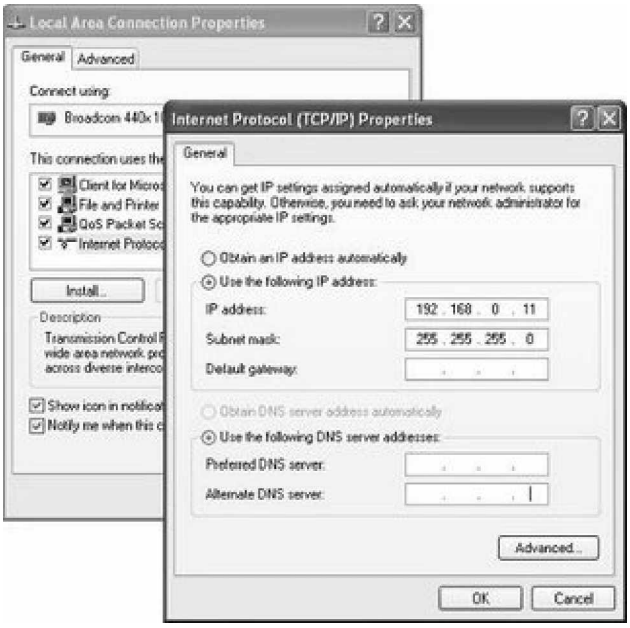
Setelah pembuatan kabel *crossover* selesai silahkan hubungkan ke kedua komputer, lalu *setting* masing-masing IP komputer dengan cara:

Buka *network connection* (dari *windows explorer* klik kanan *My Network Places -> Properties).*

Klik kanan *Local Area Connection*, lalu pilih *Properties -> Double* klik *Internet Protocol (TCP/IP).*

IP Address komputer 1 : 192.168.0.11 – *Subnet Mask* 255.255.255.0

IP Address komputer 2 : 192.168.0.22 – *Subnet Mask* 255.255.255.0



Gambar 2.5. Setting IP Protocol

Anda dapat melakukan ping terhadap komputer 2 melalui komputer 1 di DOS lewat *Start -> Run -> ketik cmd ->* lalu ketik ping 192.168.0.22

Jika komputer 2 ingin melakukan ping komputer 1 caranya sama, tinggal ganti dengan *IP address* komputer 1. Ping ini fungsinya untuk mengetahui berhasil tidaknya transfer data dari jaringan *peer to peer* yang telah dibuat tadi. Selain ping komputer 1 bisa membuka komputer 2 secara langsung di *address bar windows explorer* dengan mengetikan \\192.168.0.22 begitupun sebaliknya.

Membuat Jaringan *Peer to Peer (PC to PC)* selesai sampai di sini.

1. Merencanakan Pembuatan Jaringan

Ada banyak produk yang tersedia di toko komputer**,** namun yang perlu disiapkan adalah menentukan *piranti–piranti* yang cocok untuk *skenario* jaringan yang dibuat. Misalnya untuk menggabungkan komputer-komputer agar bisa saling berkomunikasi di tempat yang tidak

teratur atau agar komputer tetap bisa saling berkomunikasi sambil dipindahkan lokasinya dengan menggunakan *wi-fi* atau jaringan *nirkabel*. Namun jika ingin menggabungkan dua komputer PC yang berada di sutu ruangan dan hanya berjarak beberapa meter saja tentu membuat koneksi *wired* lebih efisien karena tidak perlu membeli *piranti Access Point* yang harganya lebih mahal dibandingkan hub sederhana. Aspek berikutnya adalah aspek keamanan jaringan/*security*. Yang paling aman adalah jaringan *wired* karena lebih sulit untuk di sadap *transmisi* datanya dibandingkan *wireless*. Namun *wireless* juga sebenarnya aman, namun memiliki kemungkinan disadap lebih tinggi karena data di *transmisi*kan melalui udara terbuka yang bisa di*intervesi* dengan mudah menggunakan *tool-tool* tertentu. Jaringan *wi-fi* lebih mudah dibuat karena hanya perlu menyeting *router wireless/access point* dan kartu jaringan *wireless*. Apabila ingin membuat jaringan berkabel, maka harus turut pula memasang kabel-kabel dan mengaturnya satu per satu ke komputer. Jika ada salah satu kabel yang rusak, maka harus menggantinya, hal ini dapat merepotkan dibandingkan dengan *wireless* karena tidak menggunakan kabel.

Menurut Ali Zaki dan SmitDev Community (2008) Ada beberapa hal yang perlu di perhatikan untuk membuat jaringan rumah, sebagai berikut :

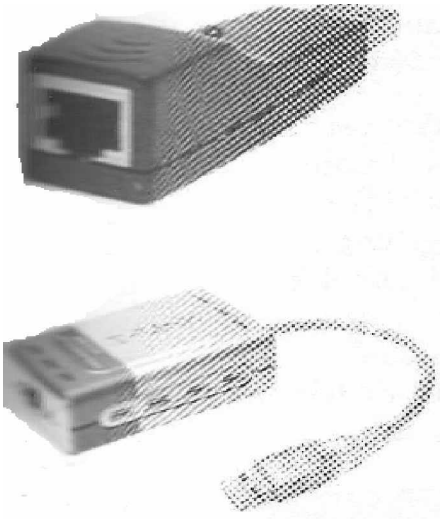
a. *Hardware* Jaringan *Wired*

Untuk membuat jaringan kabel/wired, perlu dipersiapkan beberapa *hardware* seperti berikut:

1 *Hub/switch/Router*: Untuk menghubungkan semua kabel sehingga masing-masing komputer terhubung ke *internet*.

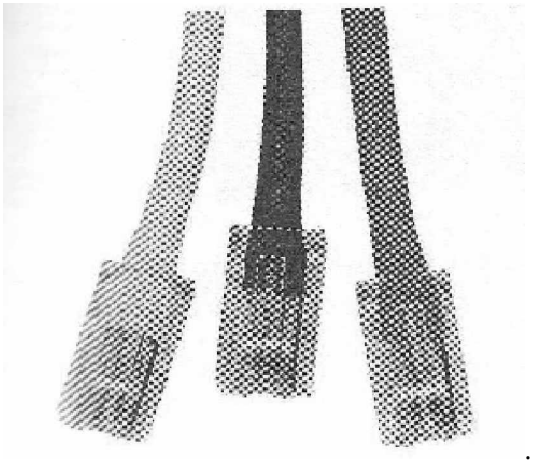
2 *Ethernet card*: Sebagai antarrmuka komputer untuk jaringan.

Gambar 2.6. *Ethernet Card PCMCIA* untuk jaringan berkabel dengan *slot RJ-45*



Gambar 2.6. *Ethernet Card USB* dengan *RJ-45*

Kabel *UTP* dengan *slot RJ-45*: Untuk menghubungkan *ethernet* ke *hub* atau bisa juga dari *ethernet* ke *ethernet* lain (untuk kasus 2 komputer langsung).

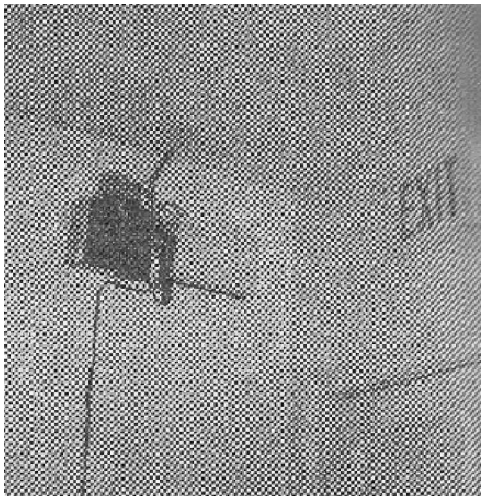


Gambar 2.7. Kabel *RJ-45* untuk membuat jaringan berkabel/*wired*

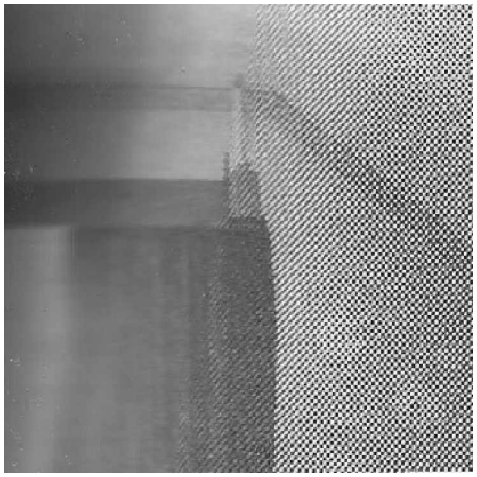
b. *Hardware* Jaringan *Wireless*

Adapun untuk jaringan *wireless*, memerlukan beberapa *hardware* seperti berikut:

1. *Router/AccessPoint*: Di *wi-fi,router* tidak hanya berfungsi untuk menghubungkan jaringan dengan internet namun juga mengandung antenna *wi-fi* (berfungsi sebagai *Access Point*) yang membagi koneksi ke komputer-komputer yang memiliki kartu jaringan *wireless*. Agar bisa memantulkan sinyal, *access point* hanya diletakkan di tempat yang terjangkau, misalnya di bagian atas.

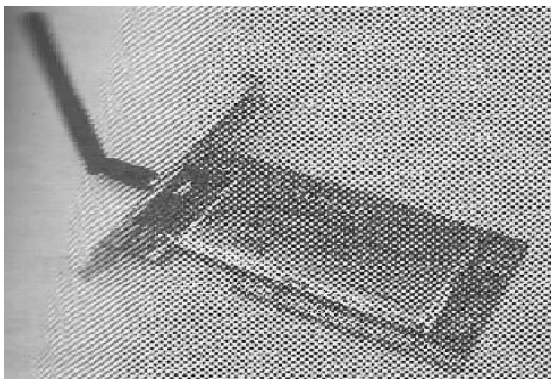


Gambar 2.8. Contoh pemasangan *access point* di bagian atas tembok dekat pintu



Gambar 2.8. Contoh pemasangan *Access point* di atas jendela

2. Kartu jaringan *wireless* yang bisa menghubungkan komputer ke *access point*. Biasanya sebuah kartu jaringan *wi-fi* bisa mengakomodasi beberapa versi *wi-fi*.



Gambar 2.9. *Wireless NIC* untuk slot PCI

3. Udara, dengan *wi-fi*, udara bisa menjadi *piranti* komunikasi antara komputer dengan *access point*.

2. Memilih Protokol Jaringan

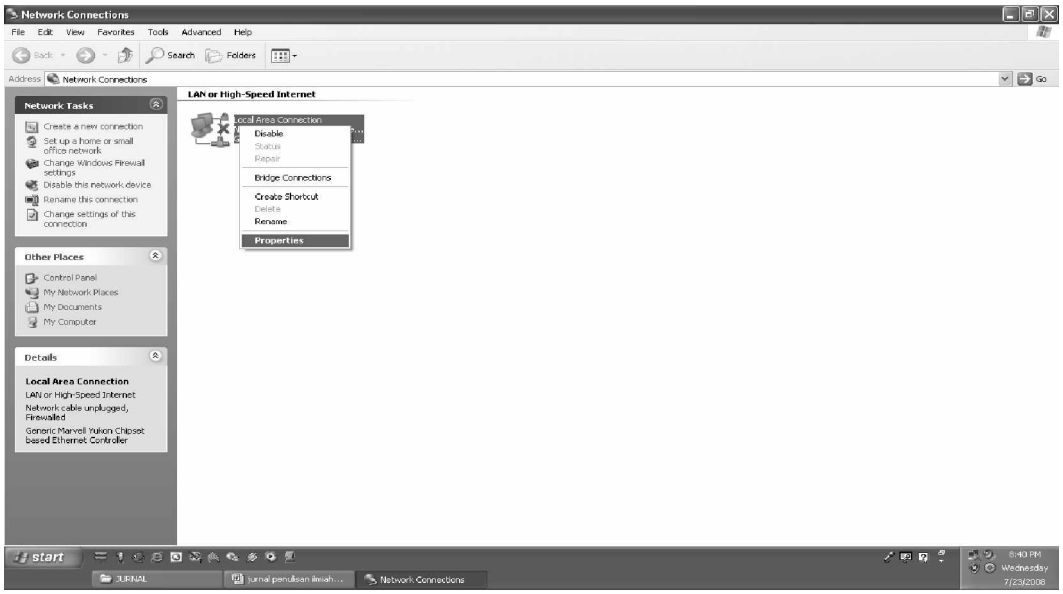
Menurut Dede Sopandi (2005) ada 3 protokol umum yang paling sering digunakan yaitu TCP/IP, NETBEUI dan IPX ketiga protokol ini dapat ditemukan dan ditambahkan pada menu *Network Neighborhood* pada *desktop windows 95/98* atau *NT*, dan *WindowsXP*.

a. *IPX (Internetwork Packet Exchange)*
Merupakan *protokol networking* dari *novell* yang menghubungkan *network* yang menggunakan *novell Netwere client* dan *server*. IPX merupakan *datagram/protokol* paket dan IPX bekerja pada layer *network* dari protokol komunikasi dan koneksi tanpa sambungan (*connectionless* = tak memerlukan koneksi yang perlu disetup sebelum paket dikirim ke tujuan).

b. *NetBEUI (NetBIOS Extended User Interface)*
Merupakan *extended version* dari *NetBIOS*, program yang memungkinkan komputer berkomunikasi did lam lingkungan *local area network*. *NetBEUI* berperformance terbagus untuk komunikasi didalam *single LAN*, karena seperti *NETBIOS* ia tidak mendukung *routing*

pesan ke *network* lain, *interface*nya perlu diadaptasikan kepada *protocol* lain seperti *IPX* atau *TCP/IP* metode yang direkomendasikan adalah meng*install NETBEUI* dan *TCP/IP* dalam setiap komputer dan menyiapkan *server* untuk menggunakan *NETBEUI* untuk komunikasi didalam *LAN* dan *TCP/IP* untuk komunikasi di luar *LAN*.

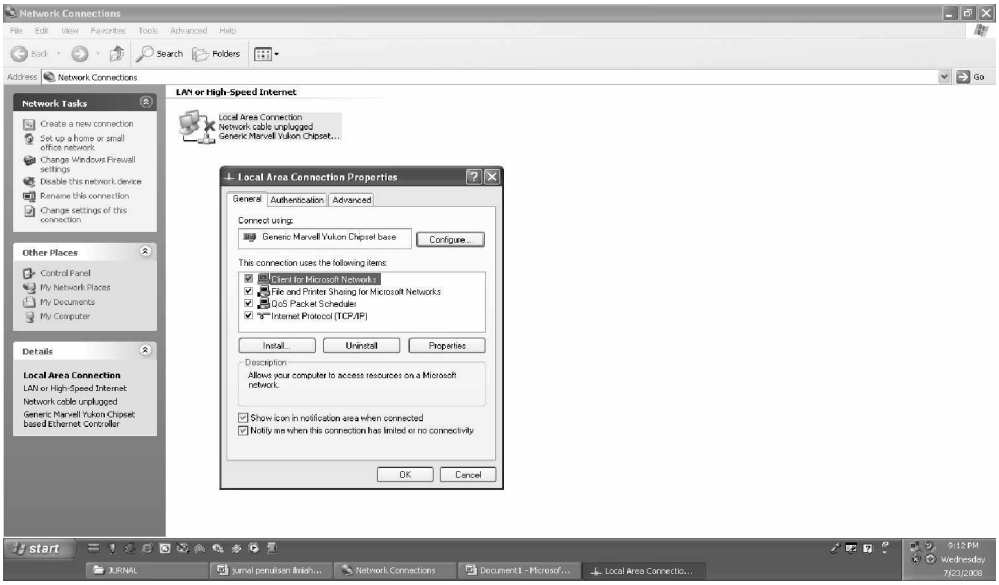
c. *TCP/IP (Transmission Control Protocol / Internet Protocol)*
TCP/IP adalah sekumpulan *protocol* yang didesain untuk melakukan fungsi-fungsi komunikasi data pada *Wide Area Network (WAN). TCP/IP* terdiri atas sekumpulan *protocol* yang masing-masing bertanggung jawab atas bagian-bagian tertentu dari komunikasi data. TCP/IP menggunakan model *client/server* dalam berkomunikasi dimana komputer *user (client)* meminta kepada komputer lain dan akan disediakan *service* tersebut oleh komputer lain itu (*server*). Internet di bentuk dari fungsi TCP/IP ini, banyak sekali protocol di internet dijalankan dengan TCP/IP protocol seperti *World Wide Web's Hypertext Transfer Protocol (HTTP), File Transfer Protocol (FTP), Telnet Simple Mail Transfer Protocol (SMTP).*
Menurut Ali Zaki dan SmitDev Community (2008) karena *protocol* yang paling unggul dan lazim dipakai adalah *TCP/IP* maka cara penggunaannya dengan mengklik kanan di ikon *My Network Places*. Di jendela *Network Connection*, kemudian memilih antar *ethernet* yang digunakan untuk menghubungkan ke jaringan komputer. Klik kanan di atas *ikon ethernet* tersebut kemudian mengklik menu *properties*.



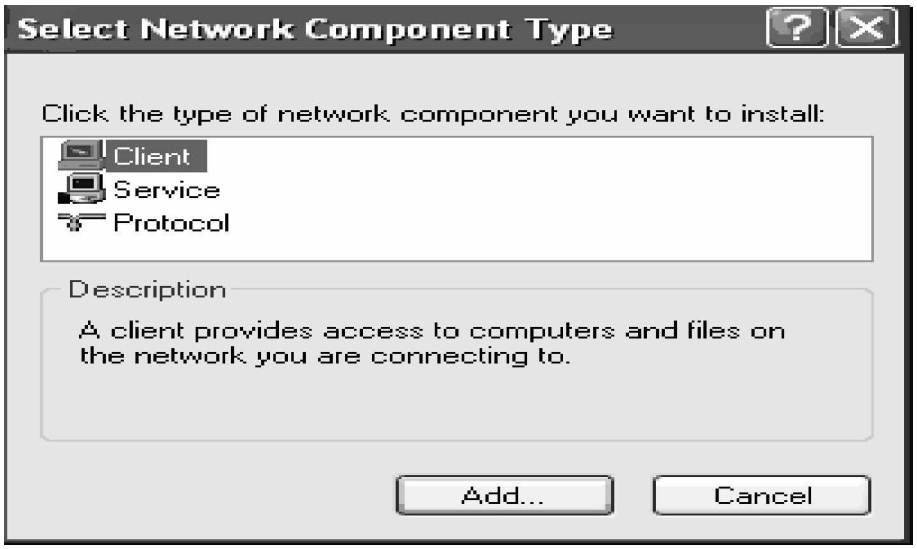
Gambar 2.10. Penentuan Properties dari sebuah *piranti ethernet card*

Ketika ditekan tombol tersebut, akan muncul jendela *Local Area Connection Properties*. Klik tombol *install*

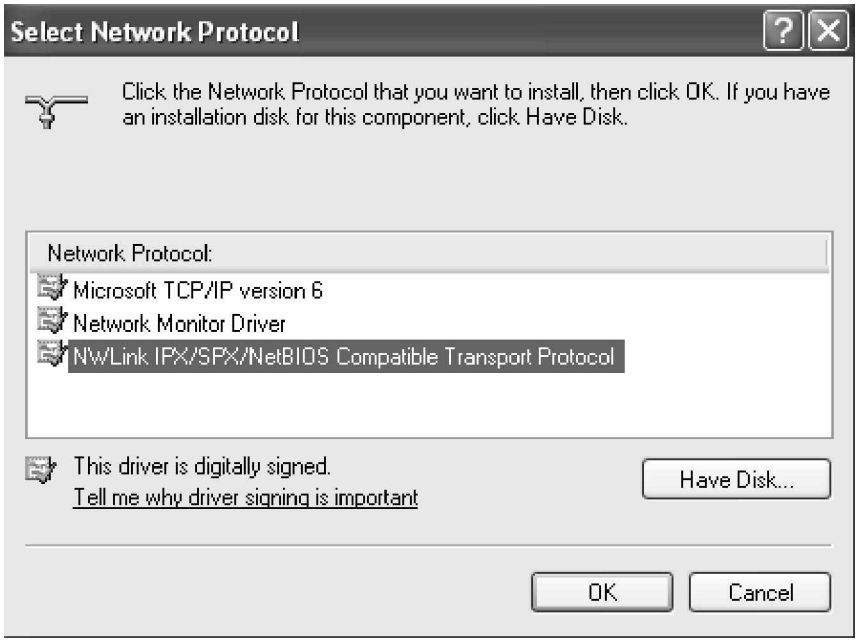
Untuk *menginstall software-software* tambahan. Pilih *Protocol* untuk *menginstal protocol* yang ingin di tambahkan. Pilih versi *protocol* yang diinginkan. Ada *IPv6, IPX/SPX*. Jika memilih *disk* untuk *menginstal aplikasi* lain, maka klik *Have Disk*. Klik OK untuk menerapkannya.



Gambar 2.11. ***Local Area Connection Properties***



Gambar 2.12. Pemilihan ***instalasi Protocol***

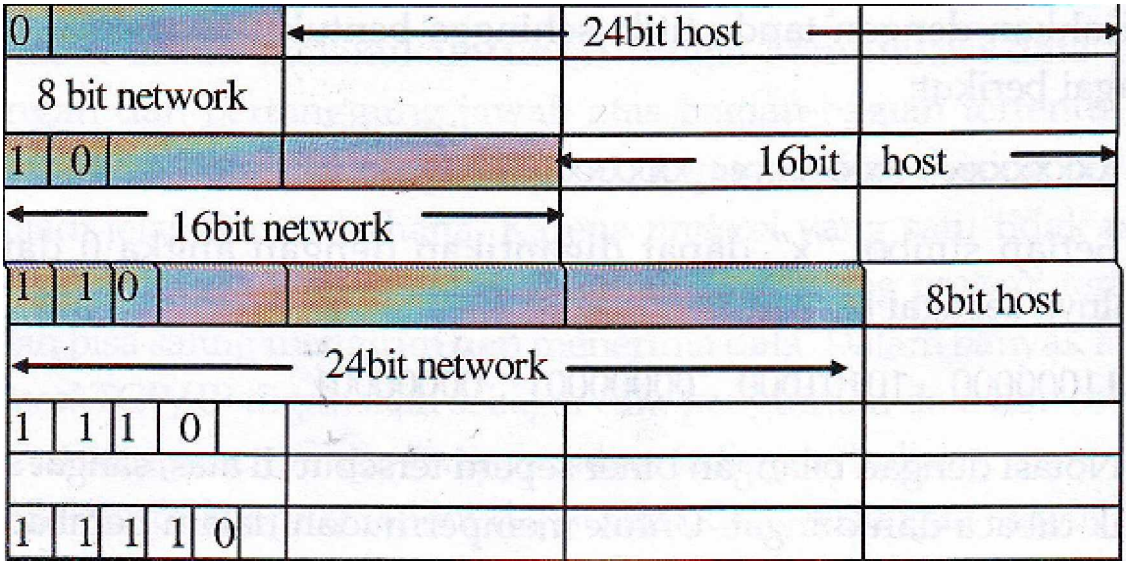


Gambar 2.13. Pemilihan *Protocol*

3. Menyetting *TCP/IP*

Untuk mempermudah proses pembagiannya, *IP address* dikelompokkan dalam kelas-kelas yang berdasarkan *network ID* dan *host ID*. *Network ID* adalah bagian dari *IP address* yang digunakan untuk menunjukkan alamat jaringan komputer yang bersangkutan berada. Sedangkan *host ID* adalah bagian dari *IP address* yang digunakan untuk menunjukkan alamat *workstation*, dan semua *host TCP/IP* lainnya dalam jaringan. Pengalamatan *IP* dibagi menjadi lima kelas, yaitu kelas A, kelas B, kelas C, kelas D, dan kelas E. Kelas-kelas ini dibagi berdasarkan beberapa *bit* awal.

Tabel 1: Kelas *IP Address*



Menurut Iwan Binanto (2007), pada gambar diatas, *IP address* kelas A mempunyai 8 *bit network ID* dan 24 *bit host ID*, artinya jaringan yang tersedia adalah 128, sedangkan jumlah *host*

per jaringan adalah 16.777.216. *IP address* kelas B mempunyai *16 bit network ID* dan 16 *bit host ID*, artinya jumlah jaringan yang tersedia adalah 16.384, sedangkan jumlah *host* per jaringan adalah 65.536. Sedangkan *IP address* kelas C mempunyai 24 *bit network ID* dan 8 *bit host ID*, artinya jumlah jaringan yang tersedia adalah 2.097.152, sedangkan jumlah *host* per jaringan adalah 256.

Tabel 2: Pembagian Jaringan *IP Address*

	Range IP	Bagian *network*	Jumlah jaringan	Bagian *host*	Jumlah *host* per jaringan
A	0.0.0.0 s.d. 127.255.255.255	1+7 bit	128	24 bit	16.777.216
B	128.0.0.0 s.d. 191.255.255.255	2+14 bit	16.384	16 bit	65.536
C	192.0.0.0 s.d. 223.255.255.255	3+21 bit	2.097.152	8 bit	256

Pengelompokkan kelas-kelas di atas, dibantu oleh sebuah komponen yang disebut dengan *netmask.*

Tabel 3: *Netmask* beserta kelas *IP address*

Kelas	Default Netmask	Jumlah IP address dalam range
A	255.0.0.0	216.77.216
B	255.255.0.0	65.536
C	255.255.255.0	256

IP address kelas E merupakan cadangan dan belum digunakan. *IP address* kelas D digunakan untuk *Multicasting*. *IP address* yang sering digunakan pada jaringan local (LAN) adalah *IP address* kelas C yang mempunyai *network ID* sepanjang 24 bit dan *host ID* sepanjang 8 bit, sehingga maksimal *host* yang dapat dihubungkan hanya 254 *host*, karena satu *IP* digunakan untuk alamat jaringan dan satu *IP* lagi untuk alamat *broadcast*. Sehingga ketika ada paket *data/program* yang dikirimkan dari satu komputer ke komputer yang lainnya, tidak secara langsung sampai ke tujuan. Hal ini disebabkan sistem akan memeriksa keterhubungannya dengan 254 komputer.

4. *Menginstall File* dan *Print Sharing*

File dan *Print Sharing* digunakan untuk membagi *resource file* dan *printer* di jaringan komputer sehingga dapat di*akses* di komputer lain di jaringan. Menurut Ali Zaki dan SmitDev Community (2008), Fasilitas *file* dan

printer sharing ini merupakan fitur yang ada secara *default* di *windows*. Namun jika ternyata belum *terinstal*, maka dapat digunakan dengan cara :

1. Buka *Control Panel* dengan mengklik tombol *Start > Control Panel.*
2. Di *Control Panel*, pilih *Network and Internet Connection*. Muncul jendela *Network and Internet Connection.*
3. Di jendela *Network and Internet Connection*, pilih *Network Connections.* Jendela *Network Connection* terbuka.
4. Di jendela *Network Connection*, klik kanan pada *Local Area Connection* dan pilih menu properties.
5. Di kotak *Area Connection Properties*, klik tombol *Instal* dan pilih *Service.*
6. Klik *OK.*
7. Di kotak *dialog Select Network Service*, pilih *File and Printer Sharing for Microsoft Networks*, kemudian klik OK. Layanan tersebut akan ditambahkan di *tab General* dari kotak *Local Area Connection Properties*. Klik *OK* untuk menutup kotak dialog *Properties*.

III. PENUTUP

3.1. kesimpulan

Dengan adanya desain pembuatan jaringan komputer berskala kecil diatas, maka pembuatan jaringan tidaklah harus berskala besar saja tetapi bisa juga di terapkan dan di praktekkan dengan berskala kecil yaitu dengan menggabungkan komputer-komputer agar bisa saling berkomunikasi di tempat yang tidak teratur atau agar komputer tetap bisa saling berkomunikasi sambil dipindahkan lokasinya dengan menggunakan *wi-fi* atau jaringan *nirkabel.*

DAFTAR PUSTAKA

Zaki Ali, Community SmitDev , 2008. *Home Networking* Membuat Jaringan Komputer untuk Rumah dan Kantor Berskala Kecil. PT.Elex Media Komputindo. Jakarta.

Binanto Iwan, 2007. Membangun Jaringan Komputer Praktis Sehari-hari. Graha Ilmu. Yogyakarta.

Sopandi Dede, 2005. *Instalasi* Dan *Konfigurasi* Jaringan Komputer. *Informatika*. Bandung.

http://www.malangkab.go.id/kabmalang/galeri – ti

http://www.situsinformasiinternet.com/2009/07/membuat-jaringan-peer-to-peer-pc-to-pc.html

PENERAPAN *INTRUSION DETECTION SYSTEM* SEBAGAI *FIREWALL* DAN SARANA UNTUK MENANGKAL PENYUSUPAN *HACKER* PADA JARINGAN LOKAL ASURANSI JIWA INHEAL INDONESIA DI JAKARTA

[1]Hendra Supendar dan [2]Tunggul Yogi Hernowo
[1]AMIK Bina Sarana Informatika
Jl. Kramat Raya No. 18 Jakarta Pusat, Indonesia
[2]Program Pascasarjana Magister Ilmu Komputer STMIK Nusa Mandiri
Jl. Salemba Raya No. 5 Jakarta Pusat (10250), Indonesia
hendrasupendar@gmail.com

Abstract
The attack on internet systems increasingly rampant, almost all of the information in the system vulnerable to attack and also generated a lot of financial losses due to these attacks, of course, as network administrators, is not an easy job to monitor the hundreds of IP in and out of clients. Install the honeypot to fool hackers also not the only way to better secure the internal regions and Demilitary Zone (DMZ). We need the help of a system that can monitor the data packet and record it and provide further information to be analyzed further. System Intrusion Detection System (IDS) can help users monitor and analyze problems in network security. In this case the IDS used a snort for windows software to monitor user activities on the system and out of the corporate network. The expected result is to know how much inconvenience caused to the system through a corporate network traffic. Data Base created using MySQL and what actions should be taken on these disorders will be the future. Testing of IDS performed on a local network of PT. Asuransi Jiwa Inhealth Indonesia *for six days. Observations were made to the directory the user can monitor the daily activities in the network and the result will be a reference in determining how much disk capacity is needed for the system is not disturbed or damaged. Observations also held with the activities of suspected hacker activities, such as sending large data packets that can make the system unstable or monitor activity in and out of IP networks that are not known or suspected. The results of this analysis is that with as many as 45 (fourty five) user disk capacity to be provided for one year is 40 GB.*

Keywords : *hacker, Intrusion Detection System, snort for windows,*

1. PENDAHULUAN

Serangan dan pencurian data yang dilakukan oleh seorang hacker untuk mengganggu sistem Komputer kita memang sudah sangat meresahkan. Banyak cara dilakukan untuk menanggulangi serangan dari *hacker* ini, mulai dari diskusi diskusi ilmiah, pembuatan peraturan tentang dunia maya dan pembelian *software* dan *hardware*
yang berharga puluhan juta rupiah, namun hal tersebut sampai saat ini belum juga dapat menjaga system secara maksimal untuk terhindar dari serangan *hacker*.

Dari pengalaman Chris Brenton seorang Instruktur dan consultan Privat SANS Institute dimana pada bulan bulan Desember tahun 2003 dia telah mengirimkan sebuah e-mail kepada *the North American Network Operators' Group* (NANOG) *mailing list*, dimana setelah mengirimkan e-mail tersebut sistem peringatan pada organisasinya menyala berkali kali, setelah diselidiki ternyata ada 16 kali upaya untuk me-relay email tersebut dari mail server-nya dari sumber IP yang sama. Andaikata pada saat itu Christ Benton tidak menggunakan IDS, maka bisa kita duga bahwa usaha *hacker* tersebut akan berhasil (Northcutt, 2004).

Mengapa setiap organisasi atau perusahaan membutuhkan pengamanan yang maksimal ? Menurut (Sherif, 2002) dalam tulisannya mengenai *intrusion treath* dikatakan bahwa dalam *network* ada lima alasan mengapa pendeteksian terhadap ancaman penyusupan sangat perlu diperhatikan secara serius.

1. *The threat is real*: sejumlah informasi keamanan pada tahun 2000-an mencatat bahwa lebih dari 70% perusahaan melaporkan serangan keamanan.
2. *Everything is on the net:* Banyak perusahaaan telah memindahkan kunci informasi dan sumber daya bisnisnya ke *internet*, dan ini telah membuka informasi sensitif perusahaan.
3. *Firewalls and VPNs are not enough:* Meskipun kebijakan *firewall* yang baik dapat meminimalkan pembukaan banyak jaringan, *hacker* mengembangkan serangan

mereka dan metode metode subversi jaringan. Teknik-teknik ini termasuk email berbasis Trojan horse, teknik *scanning* secara diam diam, dan serangan secara nyata yang mem-*bypass* kebijakan firewall dengan membuat terobosan akses diatas protocol yang diperbolehkan seperti ICMP atau DNS.

4. *The amount of new vulnerabilities is increasing*: Jumlah informasi pada jaringan yang rentan begitu meluas, banyak perusahaan sekarang menjual kepelangganan untuk mencari kerentanan , secara otomatis disesuaikan ke pada profil perusahaan dari sistem operasi dan perangkat keras jaringan. Kerentanan juga muncul di peralatan keamanan , seperti *firewall* dan bahkan peralatan IDS.

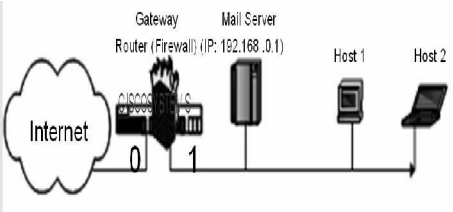
5. *Hackers are getting smarter:*
*Hacker*dapat menggunakan *scanner port* untuk mencoba terhubung ke mesin *target* pada setiap *port* dan membangun daftar potensial aktif port. *Modern scanners port* termasuk
identifikasi sistem operasi, dapat menarget kan seluruh rentang alamat IP dan bahkan mengirimkan umpan scan untuk membuatnya lebih sulit untuk target dalam mengidentifikasi sumber scanner yang sebenarnya.

2. PEMBAHASAN

Tinjauan Pustaka

A. FireWall

Firewall merupakan suatu cara atau mekanisme yang diterapkan baik terhadap hardware, software ataupun sistem itu sendiri dengan tujuan untuk melindungi, baik dengan menyaring, membatasi atau bahkan menolak suatu atau semua hubungan/kegiatan suatu segmen pada jaringan pribadi dengan jaringan luar yang bukan merupakan ruang lingkupnya. Segmen tersebut dapat merupakan sebuah workstation, server, router, atau Local Area Network/LAN (Muammar, 2004)



Gambar 1. Ilustrasi FireWall (Liu, 2004)

Karakteristik sebuah firewall adalah : (Muamar, 2004)

1. Seluruh hubungan/kegiatan dari dalam ke luar , harus melewati firewall. Hal ini dapat dilakukan dengan cara memblok/membatasi baik secara fisik semua akses terhadap jaringan Lokal, kecuali melewati firewall. Banyak sekali bentuk jaringan yang memungkinkan agar konfigurasi ini terwujud.
2. Hanya Kegiatan yang terdaftar/dikenal yang dapat melewati/melakukan hubungan, hal ini dapat dilakukan dengan mengatur policy pada konfigurasi keamanan lokal. Banyak sekali jenis firewall yang dapat dipilih sekaligus berbagai jenis *policy* yang ditawarkan.
3. Firewall itu sendiri haruslah kebal atau relatif kuat terhadap serangan/kelemahan. hal ini berarti penggunaan sistem yang dapat dipercaya dan dengan sistem yang relatif aman.

B. *Hacker*

1. Pengertian *Hacker*

Di Zaman dulu, *hacker* identik dengan pecandu komputer yang suka begadang sampai pagi, coba berbagai cara untuk mencari kelemahan keamanan sebuah sistem. Namun saat ini *hacker* adalah seseorang yang memiliki kemampuan pada komputer dan sistem jaringan, kemampuan standar yang dimiliki seorang *hacker* adalah *core programming* dan *network specialist*. *Hacker* merupakan pengguna komputer yang mampu masuk kedalam sistem komputer melalui jaringan, baik untuk keperluan *monitoring* (melihat sistem), *copying* (pengambilan/pencurian data), atau *crashing* (merusak sistem komputer) targetnya. (Chandra, 2009).

2. Klasifikasi *Hacker*

Hacker terdiri dari beberapa jenis sesuai dengan sifatnya. Baik orang itu seorang sistem *administrator* maupun *user* yang ingin membobol sistem komputer kita. Klasifikasi *hacker* antara lain sebagai berikut:

a. ***White Hats* :** merupakan *hacker* yang bekerja sebagai *system analist,system*

administrator maupun *security analist*. *White hats* bekerja dalam sistem dan memiliki kemampuan yang tinggi untuk menjaga sistem agar tetap bekerja dengan baik dan tidak diacak-acak oleh orang lain. *White Hats hackers* rata-rata memiliki sertifikat kode etik *hacker*, misalnya CEH (*Certified Ethical Hacker*) (Wikipedia, 2009)

b. ***Gray Hats* :** merupakan *hacker* yang bekerja *offensivelly* dan *defensivelly*. *Gray Hats* merupakan orang yang melakukan *attacking* terhadap sistem yang juga bekerja untuk membuat pertahanan terhadap sistem. Hacker tipe ini merupakan hacker yang membobol sistemnya untuk mendapatkan *bugs* dan lubang dari sistemnya yang kemudian mempelajari dan menutup lubang tersebut. (wikipedia, 2009)

c. ***Black Hats* :** merupakan *hacker* yang hanya bekerja sebagai *attacker* dan mengambil manfaat terhadap sistem yang diserangnya. *Black hats* merupakan *hacker* yang merusak sistem atau sering juga disebut sebagai *cracker*. Contoh aksi yang dilakukan oleh *hacker Black Hats* antara lain membobol situs perbankan, mengambil *account* (*Carding*), dsb.(wikipedia, 2008)

d. ***Suicide Hacker* :** *Hacker* yang bekerja persis seperti *Black Hats Hacker*, bersifat destruktif dan tidak peduli terhadap ancaman yang akan menimpa nya. Rata rata *suicide hacker* merupakan orang yang tidak memiliki tujuan yang jelas, hanya membobol, mengambil keuntungan, ingin terkenal dan tidak takut terhadap hukum. (wikipedia, 2008)

Melihat macam macam *hacker* diatas maka kita mungkin akan merasa bahwa apapun yang kita lakukan pasti sistem kita tidak akan pernah aman, namun kita jangan pernah menyerah dan pasrah begitu saja, mempertahankan sistem yang telah kita bangun merupakan sebuah kewajiban yang harus kita tempuh dengan jalan apa saja.

Salah satu upaya untuk mengamankan sistem adalah dengan memasang sebuah hardware atau software penangkal hacker yang akan bekerja maksimal, Kemampuan mendeteksi penyusupan secara umum disebut IDS (*Instrusion Detection System*) dan kemampuan untuk mencegah akses dikenal dengan Firewall

C. *Intrusion Detection System* (IDS)

Intrusion Detection System digunakan untuk mendeteksi aktivitas yang mencurigakan dalam sebuah sistem atau jaringan. *Intrusion* adalah aktivitas tidak sah atau tidak diinginkan yang mengganggu konfidensialitas, integritas dan atau ketersediaan dari informasi yang terdapat di sebuah sistem. IDS akan memonitor lalu lintas data pada sebuah jaringan atau mengambil data dari berkas log. IDS akan menganalisa dan dengan algoritma tertentu akan memutuskan untuk memberi peringatan kepada seorang administrator jaringan atau tidak (laing, 2000).

IDS dapat melakukan inspeksi terhadap lalu lintas komunikasi data dalam sebuah sistem atau jaringan, melakukan analisis dan mencari bukti dari percobaan penyusupan (termasuk kategori penyusupan atau tidak) dan terkadang memberikan penanganan terhadap susupan atau gangguan yang terjadi. Pendeteksian dilakukan IDS agar mem-blok gangguan jika segera dideteksi, bertindak sebagai *deterrent* (mencegah seseorang melakukan gangguan/*intrusion*), mengumpulkan informasi untuk meningkatkan keamanan.

Tipe dasar IDS menurut (Sherif, 2002)

1. *Rule-based systems* : berdasarkan atas database dari tanda penyusupan atau serangan yang telah dikenal. Jika IDS mencatat lalu lintas yang sesuai dengan database yang ada, maka langsung dikategorikan sebagai penyusupan.
2. *Adaptive systems*: mempergunakan metode yang lebih canggih. tidak hanya berdasarkan database yang ada, tetapi juga membuka kemungkinan untuk mendeteksi terhadap bentuk-bentuk penyusupan yang baru.

Bentuk yang sering digunakan untuk komputer secara umum adalah rule-based systems. pendekatan yang digunakan dalam rule-based systems ada 2, yaitu pendekatan pencegahan (preemptory) dan pendekatan reaksi (reactionary). Perbedaannya hanya masalah waktu saja. Pada pendekatan pencegahan, program pendeteksi penyusupan akan memperhatikan semua lalu lintaas jaringan. Jika ditemukan paket yang mencurigakan maka program akan melakukan tindakan yang perlu. Pada pendekatan reaksi, program pendeteksi penyusupan, hanya mengamati file log. Jika ditemukan paket yang mencurigakan program juga akan melakukan tindakan yang perlu.

Ada dua jenis IDS, yakni: (sherif,2002)

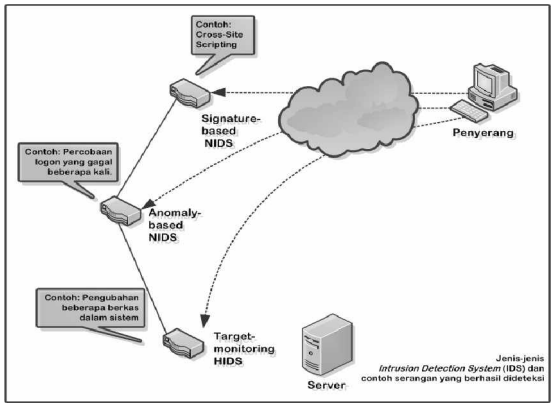
1. ***Network-based Intrusion Detection System*** (NIDS):
Network intrusion detection systems adalah jenis IDS yang bertanggung jawab untuk mendeteksi serangan yang berkaitan dengan jaringan NIDS umumnya terletak di dalam segmen jaringan penting di mana server berada atau terdapat pada "pintu masuk" jaringan. (Bueno 2002) Kelemahan NIDS adalah bahwa NIDS agak rumit diimplementasikan dalam sebuah jaringan yang menggunakan switch Ethernet, meskipun beberapa vendor switch Ethernet sekarang telah menerapkan fungsi IDS di dalam switch buatannya untuk memonitor port atau koneksi.

2. ***Host-based Intrusion Detection System*** (HIDS):
Aktivitas sebuah host jaringan individual akan dipantau apakah terjadi sebuah percobaan serangan atau penyusupan ke dalamnya atau tidak. HIDS seringnya diletakkan pada server-server kritis di jaringan, seperti halnya firewall, web server, atau server yang terkoneksi ke Internet.

Dilihat dari cara kerja dalam menganalisa apakah paket data dianggap sebagai penyusupan atau bukan maka, IDS dibagi menjadi dua: (Arvidson, 2003)

1. *Knowledgebased* atau *misuse detection*
Knowledge-based IDS dapat mengenali adanya penyusupan dengan cara menyadap paket data kemudian membandingkannya dengan database rule IDS (berisi *signature-signature* paket serangan). Jika paket data mempunyai pola yang sama dengan (setidaknya) salah satu pola di database rule IDS, maka paket tersebut dianggap sebagai serangan, dan demikian juga sebaliknya, jika paket data tersebut sama sekali tidak mempunyai pola yang sama dengan pola di database rule IDS, maka paket data tersebut dianggap bukan serangan.

2. *Behavior based* atau *anomaly based*.
Sedangkan *behavior based* (*anomaly*) dapat mendeteksi adanya penyusupan dengan mengamati adanya kejanggalan-kejanggalan pada sistem, atau adanya penyimpangan-penyimpangan dari kondisi normal, sebagai contoh ada penggunaan memori yang melonjak secara terus menerus atau ada koneksi parallel dari 1 buah IP dalam jumlah banyak dan dalam waktu yang bersamaan. Kondisi-kondisi diatas dianggap kejanggalan yang kemudian oleh IDS jenis anomaly based dianggap sebagai serangan.



Gambar 2. Jenis-jenis *intrusion detection system* dan jenis serangan yang dapat di deteksi olehnya. (http://id.wikipedia.org/wiki/Berkas:IDS.png)

3. Local Area Network

Local Area Network biasa disingkat LAN adalah jaringan komputer yang jaringannya hanya mencakup wilayah kecil; seperti jaringan komputer kampus, gedung, kantor, dalam rumah, sekolah atau yang lebih kecil. Saat ini, kebanyakan LAN berbasis pada teknologi IEEE 802.3 Ethernet menggunakan perangkat switch, yang mempunyai kecepatan transfer data 10, 100, atau 1000 Mbit/s. Selain teknologi *Ethernet*, saat ini teknologi 802.11b (atau biasa disebut Wi-fi) juga sering digunakan untuk membentuk LAN. Tempat-tempat yang menyediakan koneksi LAN dengan teknologi Wi-fi biasa disebut hotspot.

Pada sebuah LAN, setiap node atau komputer mempunyai daya komputasi sendiri, berbeda dengan konsep dump terminal. Setiap komputer juga dapat mengakses sumber daya yang ada di LAN sesuai dengan hak akses yang telah diatur. Sumber daya tersebut dapat berupa data atau perangkat seperti printer. Pada LAN, seorang pengguna juga dapat berkomunikasi dengan pengguna yang lain dengan menggunakan aplikasi yang sesuai.

Berbeda dengan Jaringan Area Luas atau *Wide Area Network* (WAN), maka LAN mempunyai karakteristik sebagai berikut :

a. Mempunyai pesat data yang lebih tinggi
b. Meliputi wilayah geografi yang lebih sempit
c. Tidak membutuhkan jalur telekomunikasi yang disewa dari operator telekomunikasi
d. Biasanya salah satu komputer di antara jaringan komputer itu akan digunakan

menjadi server yang mengatur semua sistem di dalam jaringan tersebut. (wikipedia, 2009)

4. Metode Pemasangan IDS

Menurut (Sherif, 2002), Penempatan IDS/NIDS dapat mengambil bentuk sebagai berikut :

1. NIDS dapat ditempatkan pada host yang jika tidak terjaga, misalnya, windows 98, dan tidak mampu menciptakan log yang dapat diproses oleh sistem berbasis host.

2. IDS yang paling efektif pada perimeter jaringan, seperti di kedua sisi firewall, dekat dial-up server, dan di link ke jaringan *partner*.

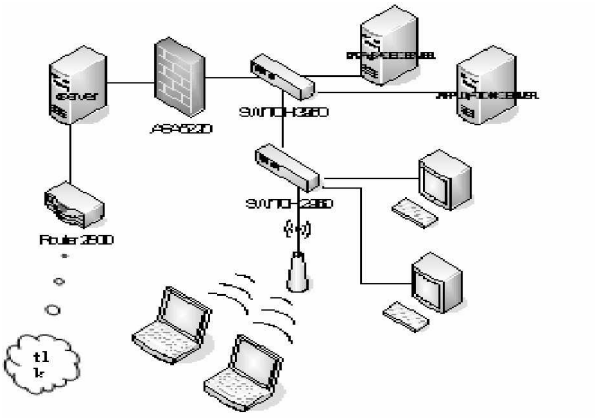
3. NIDS dapat ditempatkan pada backbone WAN perusahaan di tempat yang dapat memonitor lalu lintas paket yang mencoba untuk memasuki jaringan.

4. Untuk server farms, Salah satu solusi mungkin untuk mengisolasi kekritisan server pada segmen jaringan mereka sendiri, dan mendedikasikan NIDS khusus untuk memantau segmen tersebut.

Dalam hal ini PT AJII memilih menempatkan NIDS pada jaringan *backbone* LAN/WAN perusahaan ditempat yang dapat memonitor lalu lintas paket yang berusaha memasuki jaringan.

Untuk mendeksi serangan yang terjadi maka sebuah sistem pengamanan IDS dengan menggunakan sebuah *software* Snort di install pada jaringan lokal Perusahaan sebagai sistem peringatan jika terdapat aktivitas – aktivitas ilegal yang terjadi dalam jaringan. IDS ini dipasang pada pada sistem Windows 2003 *Server* pada jaringan. Snort yang dapat diperoleh di http://www.snort.org biasanya di sebut sebagai *Network Intrusion Detection System* (NIDS). Snort sendiri adalah Open Source yang tersedia di berbagai variasi Unix (termasuk Linux) dan juga Microsoft Windows. Database yang digunakan adalah MySQL yang diinstall pada sistem berbasis *Windows* atau sistem operasi lain yang mendukung database MySQL. Alert IDS akan disimpan pada database mysql. Untuk administrasi dan *maintenance* sistem database dibuat suatu *interface* berbasis web yang dibuat dengan bahasa pemrograman PHP. Fungsi utama dari interface ini adalah untuk mengedit atau mengupdate entry database yang dijadikan input bagi sistem yang lain. Dalam hal ini paket yang digunakan untuk file MYSQL dan PHP adalah XAMPP for *Windows*. Topologi yang digunakan adalah topologi *star* dengan kelas IP *local* adalah kelas B dan ISP menggunakan Telkom. Sebuah Router 2800 dipasang sebagai *gateaway* dan kemudian sebuah server lalu setelah itu dipasang sebuah *hardware firewall* seri ASA-5220 untuk lebih menjaga keamanan data kedalam jaringan.

Gambar Jaringan



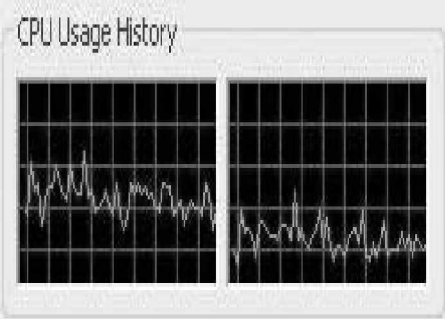
Gambar 3. Gambar Local Area Network pada Asuransi Jiwa Inhealth Jakarta

Dari gambar diatas IDS akan ditempatkan pada jaringan Perusahaan untuk megamati aktivitas para pengguna. *User* yang menggunakan akses jaringan pada PT. AJII ini kurang lebih adalah 45 *user*. Semua paket dari atau menuju klien akan melalui *Server*, sehingga untuk dapat mengamati lalu lintas paket tersebut sistem IDS ditempatkan dalam *Server* pada Router 2600. Semua paket yang masuk maupun keluar jaringan melalui *Server* akan dicerminkan ke IDS untuk kemudian dianalisis.

5. Hasil dan Pembahasan

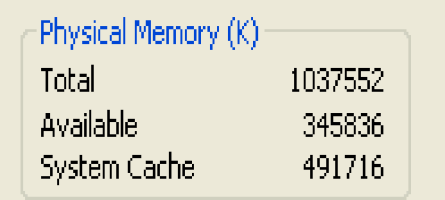
Hasil pengamatan dari implementasi selama 6 hari, dapat dikatakan bahwa sistem IDS yang di bangun mampu memonitor secara terus menerus lalu lintas data pada jaringan PT AJII dan dapat mendeteksi paket paket serangan sesuai dengan aturan aturan yang sedang aktif, kemudian memberikan peringatan kepada pengatur jaringan melalui email.

Secara teknis sistem IDS ini berjalan dengan persentase penggunaan *processor* antara 50% sampai dengan 70% dilihat dari grafik pemakaian CPU melalui *windows task manager*, seperti yang ditampilkan pada Gambar 4.



Gambar 4. CPU *Usage History*

Pemakaian RAM pada sistem ini masih menyisakan cukup banyak memori seperti yang ditunjukkan pada gambar 5.



Gambar 5. RAM *Usage*

Pengamatan dilakukan pula terhadap terhadap beberapa direktori yang berisikan berkas pencatatan total paket, jalur, kejadian dan system penyimpanan data MySQL untuk mengetahui rata rata pemakaian ruang pada harddisk tiap harinya agar pengatur jaringan dapat mengetahui seberapa besar harddisk yang diperlukan untuk menampung data selama beberapa waktu kedepan.

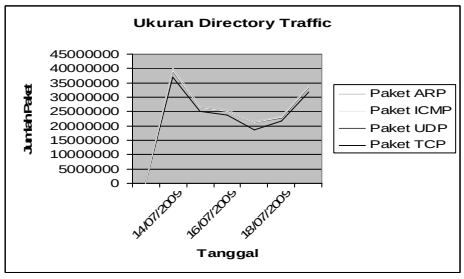
1. ***Directory Traffic* dan Total Paket**

Berikut pencatatan total paket perhari dan pertambahan ukuran pada direktori Snort/traffic (sebagai direktori pencatatan berkas total paket) seperti yang ditampilkan pada Tabel-1.

Tabel 1. Ukuran *Directory Traffic* dan Total Paket

Tanggal	Paket				Total Paket	Ukuran *Folder* Traffic (bytes)	Pertam-bahan (bytes)
	TCP	UDP	ICMP	ARP			
13/07/2009	47.493.900	2.815.582	332.691	1.147.114	51.789.288	5.179	5.179
14/07/2009	32.116.883	1.767.883	170.200	803.244	34.858.210	8.665	3.486
15/07/2009	30.586.982	1.536.660	140.082	575.761	32.839.485	11.949	3.284
16/07/2009	24.033.743	3.204.595	90.681	587.218	27.916.237	14.740	2.792
17/07/2009	27.915.202	1.491.219	87.906	747.253	30.241.580	17.764	3.024
18/07/2009	40.865.984	2.119.413	155.983	806.185	43.967.565	22.161	4.397
						Rata rata	3.694

5. Grafik Ukuran Direktori Trafik



Dari tabel dan gambar tersebut maka dapat di simpulkan bahwa pertambahan ukuran pada direktori trafik yang dicatat oleh IDS pada jaringan Lokal PT. AJII ini cukup konstan yakni antara 2792 – 5179 bytes (2-5KB) tiap harinya, dan perbandingan antara pertambahan TCP, UDP, ICMP dan ARP hampir sama setiap harinya.

2. *Directory Port*

Pengamatan kemudian dilakukan kepada direktori port yang merupakan direktori pencatatan berkas jalur setiap paket, seperti yang ditampilkan pada tabel 2 berikut ini :

Tabel 2. Directory Port

Tanggal	Ukuran Folder (Bytes)		Ukuran *Folder Port* (TCP/UDP) (*bytes*)	Pertambahan (bytes)
	TCP	UDP		
13/07/2009	99.845.682	41.334.056	141.179.738	141.179.738
14/07/2009	167.364.503	67.287.401	234.651.903	93.472.165
15/07/2009	231.667.037	89.846.286	321.513.323	86.861.420
16/07/2009	282.192.798	136.891.241	419.084.038	97.570.715
17/07/2009	340.878.492	158.783.031	499.661.523	80.577.485
18/07/2009	426.832.457	189.896.997	616.729.454	117.067.931
			Rata rata	102.788.242

Melihat tabel tersebut maka dapat disimpulkan bahwa rata-rata pertambahan ukuran pada direktori port tiap harinya sekitar 102. 788. 242 *bytes* (± 103 MB).

3. *Directory Log*

Setelah melakukan pengamatan pada direktori port, maka pengamatan dilakukan pada direktori log sebagai direktori pencatatan berkas kejadian.

Tabel 3. Pertambahan ukuran Direktori Log

Tanggal	Total Event	Ukuran *Folder Log* (TCP/UDP) (*bytes*)	Pertambahan (bytes)
13/07/2009	625	835.422	835.422
14/07/2009	2.010	3.522.181	2.686.758
15/07/2009	1.155	5.065.821	1.543.640
16/07/2009	1.774	7.438.006	2.372.185
17/07/2009	2.098	10.243.374	2.805.368
18/07/2009	697	11.175.058	931.684
		Rata rata	1.862.510

Dari tabel 3 dapat disimpulkan bahwa rata rata pertambahan ukuran pada direktori Log tiap harinya adalah sekitar 1. 862. 510 *bytes* (± 1, 9 MB)

4. Direktori XAMPP

Setelah mengamati ukuran direktori Log selanjutnya pengamatan tertuju pada besar direktori XAMPP yang merupakan direktori sistem penyimpanan data kejadian.

Tabel 4. Ukuran direktori XAMPP

Tanggal	Total Event	Total Ukuran *Folder XAMPP* (*bytes*)	Pertambahan (bytes)
12/07/2009	0	360.592.710	0
13/07/2009	625	362.252.956	1.660.245
14/07/2009	2.010	367.592.503	5.339.547
15/07/2009	1.155	370.531.641	2.939.138
16/07/2009	1.774	375.374.361	4.842.720
17/07/2009	2.098	380.949.506	5.575.145
18/07/2009	697	382.801.055	1.851.549
		Rata rata	3.172.621

Dari tabel diatas dapat kita lihat bahwa rata rata pertambahan ukuran direktori XAMPP adalah 3. 172. 621 bytes atau setara dengan 3, 2 MB

Berdasarkan hasil pengamatan dan pencatatan semua tabel diatas maka didapat data keseluruhan sebagai berikut :

Tabel 5. Pertambahan nilai seluruh direktori

No	Direktori	Jumlah maksimal dalam Mega Byte
1	***Traffic* dan Total Paket**	0,005 MB
2	***Port***	103 MB
3	***Log***	1,9 MB
4	**XAMPP**	3,2 MB
Total		108,105 MB

Berdasarkan pengamatan dan pencatatan semua tabel diatas, maka dapat diketahui bahwa rata rata ukuran hardisk yang akan terpakai setiap harinya sekitar 108,105 MB atau kita anggap saja 109 MB. Dari rata rata ini, dapat ditarik kesimpulan bahwa dengan 109 MB perhari pemakaian kapasitas hardisk untuk mendeteksi pemakaian jaringan ini maka untuk satu tahun dibutuhkan kapasitas hardisk minimal 109 MB x 360 hari = 39240 MB atau 39,240 GB setahun hanya untuk mendeteksi rata rata pemakaian kapasitas hardisk terhadap aktifitas user yang menggunakan jaringan perusahaan secara normal.

Hasil pengamatan ini juga nantinya akan menjadi acuan *administrator* jaringan untuk terus menerus melakukan pemantauan agar tidak terjadi gangguan pada sistem akibat kekurangan ruang pada *hardisk*. Bila kehabisan ruang kosong pada hardisk ini terjadi maka akan mengakibatkan sistem menjadi *hang*.

Hasil ini juga akan menjadi acuan apabila suatu waktu nanti mungkin nilai nilai dari masing masing direktori meningkat secara signifikan yang berarti bahwa akan mengakibatkan ruang yang dibutuhkan untuk menyimpan data akan semakin besar, dan bila data tersebut terlalu besar dan melebihi kapasitas hardisk yang ada maka akan mengakibatkan sistem menjadi *down*.

Sistem IDS yang ditempatkan pada server dalam jaringan juga di fungsikan untuk memantau aktifitas para pengguna jaringan PT. AJII. Sistem ini akan melakukan penangkapan paket data dari para pengguna jaringan untuk kemudian dianalisa apakah paket tersebut memiliki kriteria berbahaya atau tidak. Bila terjadi serangan atau paket tersebut merupakan paket yang berbahaya maka sistem akan memberikan *alert* berupa log file sebagai berikut :

[**] [1:499:3] ICMP Large ICMP Packet [**] [Classification:Potentially Bad Traffic] [Priority: 2] 05/09-20:15:14. 895348 10.1.4.113 -> 172.168.0.40 ICMP TTL:128 TOS:0×0 ID:6316 IpLen:20 DgmLen:65528 Type:8 Code:0 ID:512 Seq:3072 ECHO. *Alert* ini muncul ketika seseorang mencoba mengirimkan sebuah paket data yang besar dari luar sistem yang ber-IP Address 10.1.4.113 ke dalam sistem yang ber-IP Address 172.168.0.40 yang merupakan user dari jaringan PT. AJII.

Dari hasil pengamatan ini juga dapat diketahui apakah sistem yang kita bangun telah disusupi oleh *hacker* atau tidak dengan cara membandingkan hasil pengamatan yang dilakukan secara normal selama enam hari dengan aktifitas hari hari selanjutnya, bila nanti terjadi kenaikan yang signifikan pada direktori

direktori yang diamati maka sudah dipastikan bahwa ada *hacker* yang mencoba menyusup kedalam sistem jaringan.

III. PENUTUP

3.1. Kesimpulan

Berdasarkan hasil evaluasi implementasi sistem IDS pada jaringan lokal PT. Asuransi Jiwa Inhealth Indonesia, maka dapat ditarik beberapa kesimpulan sebagai berikut :

1. Sistem IDS yang dibangun dapat mengamati lalu lintas paket data dengan memberikan informasi total paket tiap protokol pada jaringan lokal PT Asuransi Jiwa Inhealth Indonesia, dengan cepat dan akurat.

2. Sistem IDS yang dibangun mampu memberikan informasi total paket yang lewat melalui tiap jalur (TCP dan UDP), sehingga memudahkan pengatur jaringan untuk mengetahui jalur jalur mana saja yang lalu lintas paketnya terlalu tinggi dari keadaan normal dan yang berkemungkinan mengganggu kinerja keseluruhan sistem pada pada jaringan ataupun ancaman ancaman lain.

3. Sistem IDS yang dibangun dapat menangkap dan menampilkan informasi yang dianggap sebagai serangan atau berbahaya sesuai dengan aturan aturan yang sedang aktif

4. Sistem IDS yang dibangun dapat berjalan selama 24 jam penuh pada jaringan tanpa mengganggu kinerja sistem lain dan juga aktivitas para pengguna jaringan PT. Asuransi Jiwa Inhealth Indonesia karena sistem IDS ini hanya mengambil paket bayangan yang dikirim maupun diterima oleh para pengguna jaringan untuk kemudian dianalisis

5. Sistem IDS yang dibangun memberikan fasilitas laporan, *export*, dan *archive* yang dapat digunakan sebagai dokumentasi dari informasi kejadian yang terjadi pada jaringan dalam kurun waktu tertentu.

6. Sistem IDS yang dibangun memungkin kan pengatur jaringan untuk menentukan kriteria aturan baru yang ingin dibuat sesuai dengan kebutuhannya, serta dapat dengan mudah mengaktifkan, menonaktifkan serta menghapus aturan aturan dari sistem penyimpanan data.

7. Dengan user sebanyak 45 user maka kapasitas yang dibutuhkan untuk menyimpan direktori direktori IDS sebesar 40 GB pertahun

3.2. Saran

1. IDS merupakan sistem pendeteksi gangguan pada jaringan. Untuk kedepan, sebaiknya sistem ini dikembangkan menjadi model Intrusion Prevention System (IPS) yang bukan hanya mendeteksi tetapi juga dapat melakukan pencegahan terhadap paket paket berbahaya yang mencoba masuk untuk merusak dan mengganggu kinerja sistem pada jaringan.

2. Sistem IDS dapat dikembangkan menjadi sistem yang tidak hanya mencatat total paket yang melewati jaringan dan mencatat pula total paket ditiap jalur namun dapat juga mencatat rincian setiap paket yang melalui jaringan untuk kemudian dianalisis lebih lanjut, dengan catatan memperhitungkan penggunaan memori pada *processor* maupun RAM serta kapasitas *harddisk* untuk sistem penyimpanan data secara otomatis.

3. Serangan dapat terdeteksi atau tidak tergantung pola serangan tersebut ada di dalam rule IDS atau tidak. Oleh karena itu, pengelola IDS harus secara rutin mengupdate rule terbaru.

4. Pemberitahuan *Alert* yang terjadi pada sistem disini hanya tampil pada monitor namun kedepan sistem *alert* dapat dikembangkan melalui pemberitahuan lewat *e-mail* maupun SMS.

5. Untuk pengembangannya, sistem IDS ini sebaiknya dilengkapi dengan metode deteksi lebih lanjut dimana sistem dapat mengenali pola serangan baru tanpa harus membandingkannya dengan pola aturan yang sudah tercatat.

DAFTAR PUSTAKA

Ajawaila, Thomas Gregory. 2003. Tutorial Membangun Snort Sebagai Intrusion Detection System Integrasi terhadap BASE dan MySQL. [*http://ilmukomputer.org/2007/02/28/mem*

bangun-snort-sebagai-intrusion-detection-system] (diakses tanggal 24 mei 2009).

Ardiyanto, Yudhi. 2008. Membangun Sistem Intrusion Detection System Yang Open Source Pada Sistem Operasi Windows. Thesis, Universitas Muhammadiyah Yogyakarta.

Ariyus, Dony. 2007. Intrusion Detection System Sistem Pendeteksi Penyusupan Pada Jaringan Komputer. Andi:Yogyakarta.

Bueno, Pedro Paulo. 2002. Understanding IDS for Linux. [*http://www.linuxjournal.com/article/5616*] (diakses tanggal 20 Juli 2009)

Chandra, Cristian A. 2009. Hackers and Their Threats: Insident Response and Protection Strategic for your company . dalam Seminar Security Attacks di Universitas Kristen Maranatha Bandung 17 Maret 2009, Diselenggarakan oleh Fakultas Teknologi Informasi, 5-10, Bandung, Fakultas Teknologi Informasi.

Hartono, Puji. 2006. Sistem Pencegahan Penyusupan pada Jaringan berbasis Snort IDS dan IPTables Firewall. [http://*www.cert.or.id/~budi/courses/security/2006/puji_report.pdf*] (diakses tanggal 16 April 2009).

Laing, Brian. 2000. Internet Security System. How To Guide Implementing a Network Based Intrusion Detection System. United Kingdom: Sovereign House 57/59 Vaster Road Reading RG18BT.

Liu, Alex X., Mohamed G. Gouda. 2004. Diverse firewall design. In Proc. of the International Conference on Dependable Systems and Networks (DSN'04), pages : 595-604. [*http://www.cse.msu.edu/~alexliu/publications/D*iver] (diakses tanggal 26 mei 2009).

Muammar, Ahmad. 2004. FireWall. Kuliah Umum Ilmu Komputer. [*http://ikc.depsos.go.id/umum/ammar-firewall.php*] (diakses tanggal 01 Juni 2009).

Northcutt, Stephen. 2004. E-mail Scums.dalam IT Ethic Hand Book, right and wrong for IT Professionals. ed. Stephen Northcutt. 125-143. Rokland : Syngress Publishing, Inc.

Sherif, Joseph S., Tommy G. Dearmond. 2002. Intrusion Detection: Systems and Models. in proc. of the Eleventh IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE'02), pages : 1-19. [*http://trs-new.jpl.nasa.gov/dspace/bitstream/2014/*] (diakses tanggal 01 Juni 2009).

Wikipedia.org. 2009. White hat hacker. [*http://id.wikipedia.org/wiki/White_hat_hacker*] (diakses tanggal 23 Mei 2009).

Wikipedia.org. 2007. Sistem Deteksi Intrusi. [*http://id.wikipedia.org/wiki/Sistem_deteksi_intrusi*] (diakses tanggal 07 Juli 2009).

Wikipedia.org. 2008. Black hat hacker. [*http://id.wikipedia.org/wiki/Black_hat_hacker*] (diakses tanggal 23 mei 2009).

Wikipedia.org. 2008. Local_Area_Network. [*http://id.wikipedia.org/wiki/Local_Area_Network*] (diakses tanggal 22 agustus 2009).

MODEL PENANGANAN *E-MAIL SPAM* DENGAN PENDEKATAN TEKNIK SMTPI

Nandang Iriadi

AMIK Bina Sarana Informatika
Jl.Oto Iskandar Dinata No. 25 Tangerang, Banten, Indonesia
nandangiriadi@yahoo.com

Abstract
The ability to use e-mail as essential to the ability to use the phone. E-mail system is very important that many people would complain if the e-mail system can not waste bekerja.E-mail (SPAM) is a problem almost all Internet users and the various efforts to control has been done. Without knowing the factors that cause the occurrence of SPAM, then this problem will still continue. This paper set forth an analysis of SPAM with SMTP Method I, to find fault in the system and the significant factors that cause other Spam.Analisis ontology of the system architecture of e-mail describing the parts and systems so that such connectedness can identify system weaknesses. Then all the significant factors associated with each other by relations of causality and proven method SMTPI messages sent by SMTP will be sent in the queue. SMTP will avoid replying to the message from the queue if connected to a remote machine .. is the use of filtering solutions that are considered quite effective from the technical point of view. filtering is essentially aimed at helping e-mail recipients to filter (select) which automatically e-mail right from spam e-mail, saving time and tenaga.Hasil analysis showed that not only technical factors such as the architecture of e-mail system and authentication process that factors in SPAM, but also other factors such as less sweeping application of security policy (security policy), and economic factors such as cost of sending an e-mail is low.

Keyword: E-mail, Spam, E-mail filter,SMTPI Methodics

I. PENDAHULUAN

Perkembangan *Internet* tumbuh dengan pesat sejak ditemukannya. Berjalan dengan itu muncullah layanan-layanan yang mendukung kegiatan manusia melalui media *internet.* Salah satu layanan yang populer sejak ditemukannya *internet* adalah *Electronic Mail (E-mail).E-Mail* yang awalnya didefinisikan sebagai surat yang berbentuk *file text* yang dikirimkan melalui *internet.* spam *(unsolicited "junk" e-mail*) atau *e-mail* sampah, merupakan problem yang mengganggu pengguna aplikasi *Internet mail. E-mail* yang berisi propaganda seringkali membuat pengguna *Internet email* merasa dirugikan, karena banyak waktu dan sumber daya yang dikerahkan untuk memilah *Spam* dan menanggu-langi kejadian ini.Produk-produk seperti *Spam Filter, blacklist provider,* dan aplikasi *e-mail khusus* yang dapat mendeteksi kehadiran *Spam* telah banyak diimplementasikan. Tetapi permasalaan *Spam* tetap belum secara tuntas dihilangkan. Untuk itu perlu dilakukan kajian yang lebih mendalam tentang Spam untuk memper-oleh faktor-faktor yang cukup signifikan sebagai penyebab *Spam.E-Mail* sekarang ini sudah berkembang lebih atraktif dengan adanya teknologi *HTML E-Mail,* sehingga *E-Mail* tidak hanya berupa tulisan saja, namun dapat disisipkan dengan gambar ataupun *file-file* lainnya. Hal ini membuat perkembangan *E-mail* awalnya dengan tujuan untuk saling berkomunikasi menjadi lebih luas lagi karena kemampuannya dapat digunakan sebagai sarana informasi dan untuk media transaksi di *internet (E-Commerce).*

Oleh karena perkembangannya serta tingkat kemudahan dalam penggunaannya *(ease of use),* beberapa orang mulai berpikir untuk menggunakan fasilitas *E-mail* untuk kegiatan *Spamming.* Pelaku *Spamming* biasanya disebut dengan *Spammer. Spammer* ini melakukan *Spamming* sehingga menghasilkan *Spam Mail. E-mail spam* sendiri didefinisikan sebagai *E-mail* yang berisi hal-hal yang tidak kita inginkan dan kadang dikirimkan oleh orang yang tidak dikenal sebelumnya. *(unsolicited commercial e-mail).* Biasanya juga disebut dengan *Bulk/Junk E-mail.*Masalah *spam* ini berdasarkan pada kenyataan bahwa biaya (*cost*) untuk mengirimkan *e-mail* ke satu orang dan 1.000

orang tidak jauh berbeda. *Barrier* untuk melakukan *mass mailing* sangat rendah. Hal ini berbeda dengan melakukan pemasaran konvensional dimana untuk mengirimkan sebuah kartu pos atau surat akan jauh berbeda untuk satu orang dan 1.000/orang. *Spam* ini tidak dikenali oleh anti-*virus* karena memang dia bukan *virus*. penyaringan terhadap *spam* harus dilakukan secara khusus. Namun mekanisme untuk melakukan penyaringan *spam* ini masih sukar karena kesulitan dalam membedakan antara *e-mail* biasa dan *e-mail* yang *spam*. Pada mulanya proses penyaringan *spam* dilakukan dengan mencari kata-kata tertentu di *e-mail* yang diterima. Kata-kata yang populer digunakan sebagai subyek dari *e-mail* antara lain *"Make money fast"*, *"viagra"*, dan seterusnya. Namun ternyata hal ini tidak efektif karena para *spammer* mengubah kata-kata tersebut menjadi kata-kata plesetan. Misalnya huruf "i" dari kata"*viagra*" diganti dengan angka "1" menjadi "*v1agra*". Hebatnya manusia adalah kita masih dapat mengerti bahwa yang dimaksudkan adalah *viagra*.Namun program komputer masih kesulitan dalam membedakan atau menyamakan kedua hal tersebut. Akibatnya jika memasukkan kata "*viagra"* ke dalam penyaringan, maka kata "*v1gra*" akan lolos dari penyaringan dan *e-mail spam* tersebut masih tetap masuk ke *mailbox* . Pendekatan berikutnya dalam melawan *spam* adalah dengan menggunakan statistik (*Bayesian*) yang menghitung kata-kata di dalam *e-mail*. Jika ada banyak kata yang merupakan kata kunci dari *spammer*, maka statistik akan menunjukan probabilitas bahwa *e-mail* tersebut merupakan *spam*. Namun lagi-lagi *spammer* lebih pintar, yaitu dengan menambahkan kata-kata yang tidak bermakna di dalam *e-mail* yang dikirimkan sehingga mengacaukan hasil statistik. (Semakin banyak kata-kata yang tidak beraturan semakin tinggi nilai *entropi* dari sinyal. semakin jauh dari label *spam*.) Jumlah e-mail *spam* ini sudah sangat banyak sehingga dapat melumpuhkan *server e-mail*. Banyak tempat yang tidak menjalankan penyaringan terhadap spam karena tidak mampu. Masalah *spam* masih menjadi masalah utama dalam sistem *e-mail* saat ini. Ada organisasi yang bernama CAUCE (*Coalition Against Unsolicited Commercial E-mail*) yang menggalang upaya-upaya untuk membendung *spam*.

II. PEMBAHASAN

Menurut Budi raharjo(2005:91)*E-mail* merupakan aplikasi yang paling utama di jaringan *Internet*. Hampir setiap orang yang menggunakan Internet memiliki alamat *e-mail*. Saat iniakan aneh jika anda tidak memiliki alamat *e-mail*. Kemampuan kemampuan menggunakan *e-mai*l sama esensialnya dengan kemampuan menggunakan telepon .Sistem *e-mail* sudah sangat pentingnya sehingga banyak orang akan mengeluh jika sistem *e-mail* tidak dapat bekerja. Bahkan banyak bisnis yang dilakukan dengan menggunakan *e-mail*. Dapat dibayangkan jika sistem e-mail tidak dapat bekerja dalam waktu yang lama.Sebelum mendiskusikan permasalahan *e-mail*, ada baiknya kita kenali dulu sistem *e-mail*. *E-mail* merupakan salah satu media komunikasi utama dalam lingkungan internet. *Electronic mail* adalah salah satu sarana komunikasi yang cukup handal, perbandingan-nya dengan mail adalah waktu pengirimannya yang sangat cepat. *Electronic mail* atau dising-kat *e-mail* bukanlah pelayanan "*end to end*", karena mesin pengirim dan penerima tidak perlu berkomunikasi secara langsung. Proses penyampaian *electronic mail* dapat dianalogikan dengan penyampaian surat oleh Kantor Pos dan Giro. Kegagalan pengiriman *e-mail* pada umumnya disebabkan karena kesalahan menulis alamat dapat juga disebabkan oleh mesin penerima atau sebagian jaringan mengalami ganguan, tetapi biasanya jaringan mencoba beberapa kali sebelum gagal..Proses ini disebut *"store and forward"*. Alamat yang dituju harus ditentukan dan kemudian surat tadi diletakkan di kotak pos, kemudian mobil pos akan mengambil surat tersebut untuk dibawa ke kantor pos terdekat, dari kantor pos asal kemudian surat tadi dikirim ke kantor pos terdekat dengan alamat tujuan dan akhirnya dari kantor pos tadi surat tersebut dikirim ke kotak pos tujuan.Seiring dengan perkembangan tersebut, muncul juga penyalahgunaan dari fungsi *Email* yang ada. Beberapa pihak mulai menggunakan media ini untuk mengirimkan *Spam Email* yang merugikan dan tidak dikehendaki oleh penerima. Hal tersebut menimbulkan banyak kerugiandi kalangan komunitas internet. Menurut Budi raharjo(2005:92) Sistem *e-mail* terdiri dari dua komponen utama, yaitu

1. *Mail User Agent* (MUA)

MUA merupakan komponen yang digunakan oleh pengguna *e-mail*.Biasanya disebut program mail. Contoh MUA adalah *Eudora*.

2. *Mail Tranfer Agent* (MTA).

Netscape, Outlook, Pegasus, Thunderbird, pine,

mutt, elm, mail, dan masih banyak lainnya lagi. MUA digunakan untuk menuliskan *e-mail* seperti halnya mesin ketik digunakan untuk menulis surat jaman dahulu.MTA merupakan program yang sesungguhnya mengantar *e-mail*. Biasanya dikenal dengan istilah *mailer*. MTA ini biasanya bukan urusan pengguna, akan tetapi merupakan urusan dari *administrator*. Contoh MTA antara lain *postfix, qmail, sendmail, exchange, MDaemon, Mercury,* dan seterusnya.

A. E-mail Spam

Menurut Budi Raharjo(2005:99) *Spam* adalah didefinisikan sebagai "*unsolicited e-mail*", yaitu *e-mail* yang tidak kita harapkan. *Spam* ini berupa *e-mail* yang dikirimkan ke banyak orang. Biasanya isi dari *e-mail* ini adalah promosi.*Spam* adalah pengiriman sejumlah *message* yang sama yang memaksa orang lain yang tidak menginginkan adanya *message* itu mau tidak mau harus menerimanya. *e-mail* merupakan salah satu teknologi yang sangat membantu manusia untuk berkorespondensi satu sama lain. Hal ini disebabkan e-mail tidak mengenal batasan waktu, tempat dan biaya yang amat murah ketika seseorang ingin melakukan korespondensi dengan kenalannya di seluruh dunia selama dapat mengakses jaringan internet. *E-mail juga* telah mengubah cara perusahaan menjalankan bisnis mereka khususnya berinteraksi (surat menyurat) dengan pelanggan mereka. Namun ada beberapa hal yang perlu di perhatikan mengenai keamanan *e-mail.* Selama *e-mail* masih menggunakan jaringan internet sebagai media penghantarnya, maka *e-mail* juga sangat rentan terhadap celah keamanan seperti yang terjadi pada jaringan *internet. Spam* merupakan salah satu celah keamanan dalam *e-mail* sehingga di perlukan suatu mekanisme pengamanan *e-mail* yang efektif untuk mengurangi celah keamanan tersebut. *Spam* selalu menuliskan bahwa mereka akan menghapus nama-nama orang yang tidak ingin berada didalam daftar pengiriman spam tersebut, tapi hamper tidak dilakukan *Spammer* juga tau bahwa para pengguna *internet* tidak ingin menerima pesan dari mereka, sehingga mereka menggunakan alamat palsu di message yang mereka sehingga mereka tidak perlu menanggung biaya penerimaan balasan dari orang-orang yang mereka kirimkan pesan tersebut, dapat juga mereka memanfaatkan masa "trial" dari suatu ISP. Bahkan walau kita mengabaikan mengapa kita harus melakukan sesuatu untuk keluar dari list yang kita tidak pernah ikut, hal ini menjadi sangat tidak mungkin bila volume dari spam itu menjadi lebih besar. Sekarang ini, kebanyakan dari kita hanya mendapat beberapa spam per-harinya. Tetapi jika 1/10 dari 1% user internet memutuskan untuk membuang spam tersebut dengan rate menengah yaitu 100.000 per-hari, rate tersebut dapat dengan mudah dicapai dengan dial-up account dan sebuah PC. Maka tiap orang akan menerima 100 spam perharinya. Jika 1 % user melakukan spamming dengan rate tersebut, maka kita semua akan memperoleh 1000 spam perharinya. Apakah realistis jika kita realistis jika kita meminta orang untuk men-delete 100 message di inbox-nya perharinya? Tentu sulit. Dapat dibayangkan jika spam semakin berkembang.Kebanyakan *spam* yang ada di *internet* adalah berupa iklan, promosi dari suatu produk. Peningkatan jumlah spammers, seperti *Quantum Communciations*, mengirimkan sebagian atau bahkan seluruh mail miliknya melalui intermediate system, untuk menghindari pem-block-an dari suatu system untuk spammers. Hal ini akan memenuhi network dan disk dari intermediate system dengan message spam yang tidak diinginkan, akan menghabiskan waktu dari pengelola system.Kebanyakan spammers menggunakan teknik "*hit and run*" spamming dimana mereka menggunakan account dari sebuah *provider* selama beberapa hari (untuk *trial*) untuk mengirimakan ribuan dan kemudian meninggalkan *account* tersebut, sehingga akan meninggalkan masalah baru bagi para provider. Kebanyakan spammer telah melakukan hal ini puluhan kali, memasa para provider untuk meluangkan waktu ekstar untuk membereskan masalah ini.*Spam* ini sangat menguntungkan bagi pengirim, karena hanya membutuhkan biaya yang kecil, sementara tidak sebaliknya jika kita memandang dari sisi penerima. Menurut Anjik dan Rianto(2008:173) *Spam* adalah suatu *e-mail* yang membawa pesan-pesan yang sifatnya komersial.*spam* pertama terjadi pada bulan Mei 1978 dijaringan *ARPANET* sebelum *internet* terbatas dalam lingkup militer dan akadenis di prakasai oleh seorang pekerja *Digital Equipment Corporation*(DEC).*Spam* dikirimkan oleh pembuat iklan dengan biaya operasi yang sangat rendah, karena *spam* ini tidak memerlukan senarai (*mailing list*) untuk mencapai para pelanggan-pelanggan yang diinginkan. Sebagai akibatnya banyak pihak yang dirugikan. Selain pengguna *Internet* itu sendiri, ISP (Penyelenggara Jasa *Internet* atau *Internet Service Provider*), dan masyarakat umum juga merasa tidak nyaman. Karena biasanya sangat

mengganggu dan kadang-kadang membohongi, berita *spam* termasuk dalam kegiatan melanggar hukum dan merupakan perbuatan pidana yang bisa ditindak melalui undang-undang *Internet*.

B. Tipe Spam

Menurut Iron Port(2004).Ada dua tipe spam dengan akibatnya masing-masing pada pengguna internet, yaitu :

1. Cancellable Usenet spam

Merupakan sebuah pesan yang dikirmkan kepada 20 atau bahkan lebih pengguna *Usenet newsgroup*, pesan ini kebanyakan tidak berguna bagi anggota *Usenet newsgroups*. *Usenet spam* ini biasanya ditujukan pada orang-orang yang disebu *"lurkers"* yaitu orang-orang yang membaca newsgroup tetapi jarang mem-posting atau memberikan alamat *e-mail* nya. *Usenet spam* ini membebani user dengan mengirimkan posting-posting yang tidak perlu. Selain itu *Usenet spam* ini juga akan membatasi *system administrator* ataupun pemilik untuk mengatur hal-hal yang penting yang seharusnya dimasukkan dalam sistem miliknya

2.* Target *E-mail spam

Target dari *e-mail spam* ada *user* individu dengan *e-mail message* secara langsung. *List* dari *e-mail spam* ini biasanya didapat dari Usenet posting, mencuri *mailing list* tertentu dari internet, atau mencari alamat-alamat dari *web*. *E-mail spam* ini biasanya membebani user untuk mengeluarkan uang lebih banyak karena menerimanya.Bentuk lain dari *e-mail spam* adalah mengirimkan spam tersebut kedalam suatu *mailing list* (baik publik maupun *private*), karena kebanyakan *mailing list* membatasi aktivitas dari para penggunanya, *spammers* akan menggunakan alat otomatis untuk memasukkan *spam* kesebanyak-banyaknya ke *mailing list*, sehingga mereka bisa mendapatkan alamat sebanyak-banyaknya ataupun dengan menggunakan *mailing list* sebagai target langsung pengiriman *spam*

C. **Penyaringan *Spam***

penyaringan merupakan penyelesaian utama yang dianggap cukup efektif dari segi teknis. penyaringan pada intinya bertujuan membantu penerima *e-mail* untuk menyaring (memilih) secara otomatis mana *e-mail* yang benar dan mana spam, sehingga menghemat waktu dan tenaga. Sejak maraknya *spam*, telah berkembang banyak solusi penyaringan. Dari yang sederhana hingga menggunakan algoritma kompleks. Dari yang bersifat personal hingga bersama-sama. Dari yang gratis sampai jasa komersial oleh pihak ketiga. Jika kita memperoleh *e-mail* pertama kalinya dari seseorang, maka *challenge-response filter* akan mengirim *e-mail* kembali ke alamat pengirim tersebut dan memerintahkannya untuk meng-akses alamat *web* tertentu dan mengisi suatu *form* sebelum *e-mail* yang ia kirim dapat kita terima. Dengan cara seperti ini, kita dapat menyaring *spam* dengan akurat. Karena hanya pengirim yang benar-benar berkepentingan dengan kita yang akan melaksanakan prosedur tersebut. Tetapi metode ini dapat dikatakan kasa, karena membuat orang lain melakukan pekerjaan ekstra untuk mengirim *e-mail* kepada kita. Selain itu kekurangan metode ini adalah *legitimate e-mail* dapat hilang atau terlambat sampai, karena pengirimnya tidak mengetahui bahwa ia harus melakukan suatu prosedur dari *challenge-response filter* agar *e-mail* nya dapat diterima. Kekurangan yang lain adalah karena penyaringan ini hanya menyeleksi *e-mail* berdasarkan alamat pengirimnya, maka spammers yang melakukan *spoofing* akan mampu menaklukkan penyaringan ini. Sehingga penyaringan ini tidak terlalu efektif untuk memblok *e-mail spam*. Cara yang dapat dilakukan adalah dengan mengkombinasikan filter ini dengan penyaringan secara statistik, yaitu *e-mail* yang dikategorikan sebagai spam oleh penyaringan secara statisitk, merubah kembali oleh dengan adanya perubahan penyaringan ini. Dengan cara seperti ini, keakuratan Bayesian penyaringan akan bertambah, dan merespon penyaringan juga dapat digunakan dengan efektif *spam*,penyaringan hanya akan membant meringankan masalah yang ada. Di lain hal filtering juga dapat mendatangkan *spam-spam* baru. Hal ini bisa terjadi karena pihak penjual penyaringan *spam* komersial dengan sengaja memasukkan nama clien nya sebagai daftar daftar *spam*. Maka pihak pengelola *server* (*client*) yang tidak ingin terganggu oleh *spam* terpaksa membeli penyaringan dari pihak komersial. penyaringan dapat memilah *e-mail* yang benar dengan *spam* tapi penyaringan tidak mampu mencegah masuknya *spam* ke dalam jaringan. Untuk itu dibutuhkan suatu cara untuk menahan *spam* yang biasa disebut *realtime black hole*. Cara ini untuk mementahkan *spam* yang akan masuk kedalam jaringan,*spam-spam* tersebut datang dari pihak (mesin) lain. Daftar mesin-mesin yang mengirimkan *spam* ini akan terus di perbaharui oleh suatu organisasi,daftar ini dapat dimanfaatkan untuk menolak *e-mail* atau apapun bentuknya yang datang dari mesin yang terdaftar sebagai pengirim *spam*. Menurut Neal Krawetz(2004)Penyaring berbasis anti

spam mempunyai pendekatan yang signifikan. keterbatasannya adalah sebagai berikut:

a. **Mengkombinasikan penyaringan**

Pengirim spam dan bulk-mailing aplikasi tidak statis , mereka cepat beradaptasi sekitar filter. Misalnya, untuk counter daftar kata, pengirim *spam* mengacak ejaan kata (*"viagra", "V1agra", "\ / iaagra").* karakter yang berbeda di setiap e-mail diciptakan untuk *bypassing hash filter*. Dan yang sedang populer penyaringan secara statistik dengan masuknya kata-kata dan kalimat acak. Penyaringan *spam* yang paling efektif hanya untuk beberapa bulan . Dalam rangka untuk menjaga kelangsungan anti *spam,* sistem penyaring menetapkan aturan harus terus diperbarui - biasanya pada harian atau mingguan. Penyaringan adalah penyelesaian terutama dari segi teknis. penyaringan pada intinya bertujuan membantu penerima *e-mail* untuk memilah-milah secara otomatis mana *e-mail* yang benar dan mana *spam*, sehingga menghemat waktu dan tenaga. Sejak maraknya *spam*, telah berkembang banyak solusi penyaringan. Dari yang sederhana hingga menggunakan algoritma kompleks. Dari yang bersifat personal hingga secara bersama-sama. Dari yang gratis sampai jasa komersial oleh pihak ketiga.Saat ini teknologi penyaringan sudah cukup memuaskan. Solusi seperti *Spam Assassin* misalnya yang menjadi favorit banyak *sysadmin/user* menggunakan berbagai cara untuk mengidentifikasi *spam*. Mulai dari deteksi *header*, pencarian kata-kata yang umum ada di *spam*, hingga integrasi dengan sistem penyaringan lain. murni mengandalkan analisis konten sebab, Inti dari spam adalah pesannya. Jadi yang harus kita usahakan adalah mengenali *spam* dari pesan.Kualitas penyaringan ditentukan dari *rendahnya* pesan biasa salah terdeteksi sebagai *spam* dan tingginya kebenaran. Dengan penyaringan yang ada sekarang, telah dimungkinkan mencapai akurasi di atas 95% kebenaran dan kesalahan mendekati 0%. Bahkan banyak penyaringan telah memiliki kemampuan untuk melakukan *autoreporting:* manakala *spam* ditemukan, langsung dilaporkan atau ditambahkan ke dalam basisdata untuk membantu proses penanganan *spam* lainnya. penyaringan hanyalah satu sisi dari solusi berbasiskan teknis. penyaringan tidak menyelesaikan akar permasalahan, hanya membantu meringankan beban penerima dalam menyortir *e-mail*. Meskipun disaring namun jumlah total *spam* yang sebetulnya masuk setiap hari terus bertambah. Ada sisi lain juga masalah penyaringan ini, yaitu adanya pihak-pihak yang bermain di air keruh. Ada beberapa penjual solusi penyaringan komersial yang dengan sengaja memasuk-masukkan calon klien ke dalam berbagai daftar spam. Dengan tujuan membuat klien terpaksa membutuhkan penyaringan. Jadi di sisi ini maraknya penyaringan justru akan mendorong meningkatkan jumlah *spam*.

b. Berpikiran Positif

Yang lebih efektif penyaringan *spam*, semakin tinggi kemungkinan *e-mail* sebagai *spam*. Sebagai contoh, *e-mail* yang berisi kata "*viagra*" Hampir pasti akan ditandai sebagai spam tanpa isi. Sebaliknya, penyaringan *spam* yang hampir tidak menghasilkan pikiran positif yang mungkin untuk menghasilkan sejumlah pikiran negative mengenai *spam*.

c. Meninjau Kembali Penyaringan Spam

Karena kemungkinan pesan palsu, yang ditandai sebagai s*pam* biasanya tidak segera dingenai hapus. Sebaliknya, pesan akan disimpan dalam kotak masuk untuk diperiksa. ini berarti bahwa pengguna masih harus melihat *spam*, meskipun hanya oleh subjek, karena mereka mencari *e-mai*l. Pada dasarnya, penyaring hanya membantu penyortiran *e-mai* masuk.

Menurut Brian Bagnall(2000:43) Penyaringan teks pada *e-mai l* terdiri dari tiga bagian:

1. **Menambahkan alamat *e-mail* ke daftar pengirim.**

Ketika kamu mendapat satu *e-mail* yang kamu pengaruhi seperti *junk e-mail* yang dapat menambahkan alamat *e-mail* ke Konten atau daftar pengirim melalui Aksi *junk e-mail* Tambahkan ke daftar Pengirim *junk e-mail* Menambahkan ke Dewasa Daftar Pengirim konten. Waktu berikutnya kamu mendapat satu *e-mail dari* ini pengirim, aksi ditetapkan berlaku bagi ini.

2. **Tambahkan alamat *e-mail* untuk daftar eksepsi.**

Satu *e-mail* mungkin diidentifikasi sebagai jong, tapi kamu tidak mempengaruhi alamat pengirim ini seperti

demikian. menempatkan pengirim ini alamat *e-mail* pada daftar eksepsi. Aktifkan Alat Pemandu Ketentuan dan lihat aturan memanggil Eksepsi. bagian dari Ketentuan dapat mengedit nilai daftar eksepsi dengan memilih . Satu mengedit tulisan yang memperbolehkan kamu untuk memelihara daftar alamat *e-mail* yang mencegah *e-mail* berasal dari pengirim ini untuk disampaikan ke penyaringan *e-mail.*

3. **Perbaharui penyaringan konten**

Sesuatu akan mengasumsikan bahwa penyaringan *e-mail* mampu untuk perubahan, tapi dapat menciptakan penyaringan *e-mail* sendiri berlandaskan berkas tulisan satu praktek lebih baik untuk mencek Situs untuk pembaharuan, atau cari cara di *internet* untuk penyaringan dari pihak ketiga.

D. Metode Penyaringan

Menurut Jonathan A dziarski (2005:40)terdapat beberapa metode penyaringan yang dapat digunakan untuk mencegah spam diantaranya :

a. Keyword filtering

Metode ini merupakan *Application Layer Filtering* (ALF). Dengan metode ini, *spam* di-blok berdasarkan kata-kata tertentu yang sering dituliskan pada *spam-mail*

b. Signature –Based Filtering

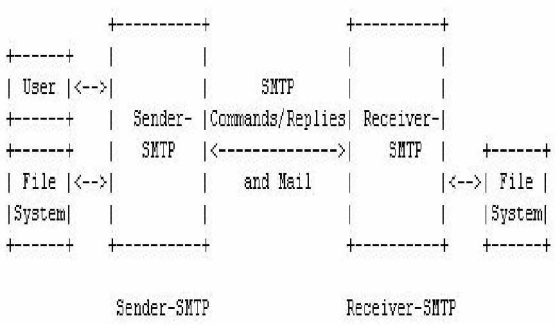
Metode ini akan membandingkan *e-mail* yang datang dengan *spam-mail* yang telah diketahui. Hal ini dilakukan dengan membuat beberapa alamat *e-mail* palsu. *Spam-mail* yang biasanya dikirim ke beratus-ratus alamat *e-mail* juga akan dikirim ke alamat-alamat palsu ini. Sehingga dengan membuat daftar alamat-alamat mana saja yang mengirim *e-mail* ke alamat palsu ini, *e-mail spam* dapat diblok. Salah satu cara untuk menunjukkan bahwa dua buah e-mail sama dilakukan dengan memberikan "*signature*" pada setiap *e-mail.* Metode untuk memberikan s*ignature* antara lain dengan memberikan angka untuk setiap huruf, lalu semua angka tersebut dijumlahkan. Sehingga setiap *e-mail* akan memiliki "*signature*" yang berbeda. Dalam hal ini, dua e-mail yang memiliki signature yang sama, dan dikirim ke beberapa alamat dapat dikategorikan sebagai *e-mail spam.* Cara inilah yang diterapkan pada *signature-based filtering*. Tetapi metode penyaringan ini sangat mudah dikalahkan oleh *spammers*. Cukup dengan menambahkan sembarang karakter yang berbeda pada setiap *copy spam-mail,* akan membuat memperbanyak *e-mail spaml* itu memiliki signature yang berbeda. Sehingga metode ini tidak terlalu efektif untuk menyaring *spam*. *Spammer* selalu mencari cara untuk mengelabui *Spam Filter* dengan melihat cara yang dilakukan oleh *Spam Filter* yang digunakan. Selain itu juga digunakan teknik yang bersifat *rule* untuk mengurangi tingkat kesalahan *False Positive* dari metode *statistic.*

c. Bayesian (Statistical) Filtering

Metode Penyaringan secara statistik merupakan metode anti penyaringan *spam* yang terbaru. Metode ini mengenali *spam* berdasarkan kata-kata (token) yang terkandung pada sebuah e-mail. Metode penyaringan ini pertama kali perlu pelatihan dengan menggunakan dua koleksi *e-mail,* satu koleksi merupakan *e-mail spam*, dan koleksi yang lain merupakan *e-mail* yang secara sah. Dengan cara seperti ini, pada setiap *e-mail* baru yang diterima, penyaringan secara statistik dapat memperkirakan probabilitas *spam* berdasarkan kata-kata yang sering muncul di koleksi *e-mail spam* atau di koleksi *e-mail* Panyaraingan secara statistik efektif untuk memblok *spam* karena penyaringan ini dapat secara otomatis mengkategorikan *e-mail spam* atau *legitimate e- mail.* Kelemahan dari *Bayesian chain rule* ini adalah tiap kata diasumsikan terpisah dan tidak tergantung satu sama lain. Padahal dalam menganalisis suatu teks, setiap kata saling berhubungan satu dengan yang lain.

***E. Simple Mail Transfer Protocol* (SMTP)**

Menurut Jonathan B..Postel(1982) tujuan *Simple Mail Transfer Protocol* (SMTP) adalah untuk mentransfer *mail* yang dapat dipercaya secara efisien, SMTP ini bebas terhadap subsistem transmisi dan hanya membutuhkan stream kanal data yang dapat dipercaya.*Fitur* yang penting dari SMTP ini adalah kemampuannya untuk merelay e-mail melalui berbagai macam *transport* servis, karena Mail adalah aplikasi atau penggunaan dari komunikasi interproses



Gambar 1 Model SMTP(Jonathan 1982)

Cara kerja dari model SMTP ini adalah sebagai berikut : Jika ada permintaan e-mail dari pengguna, maka pengiriman SMTP akan menyediakan kanal transmisi 2 arah kepada pengiriman SMTP. Pengiriman SMTP dapat menentukan tujuan atau meneruskannya lagi. *Command* SMTP dikeluarkan oleh sender SMTP dan dikirmkan kepada pengiriman SMTP, dan SMTP replies pada arah sebaliknya.Apablika

kanal transmisi sudah diberikan, sender SMTP akan mengirimkan *e-mail* command yang menandakan pengiriman *e-mail.* Bila penerima SMTP dapat menerima *e-mail* maka ia akan mengirim dengan ok kirim, kemudian SMTP akan mengirimkan RCPT *command* yang memberitahukan penerima dari e-mail tersebut, jika tidak maka pengiriman SMTP akan menolaknya.SMTP menggunakan beberapa *spoo*l dan *queue*. Pesan yang dikirim oleh SMTP akan dikirimkan dalam *queue.* SMTP akan menghindari membalas pesan dari *queue* jika dihubung-kan ke *remote machine. Jika* pesan tidak dapat dibalas dengan waktu yang telah ditentukan maka pesan akan dikembalikan ke pengirim atau dipindahkan. Interaksi antara *Message User Agent* dan ke *Message Transfer Agent* hingga diterima oleh Penerima Jika lebih dari satu penerima pada lokasi penerima dikenali, beberapa RCPT akan dikirim tetapi pesan itu sendiri yang hanya dikirim sekali. Setelah masing-masing RCPT akan terdapat beberapa pengenalan. Perintah data diikuti oleh jalur -jalur pesan sampai dengan periode tunggal pada line itu sendiri yang mengenali akhir dari pesan. Keter-hubungan akan ditutup dengan perintah *quit.*Alamat pengirim dan penerima menggunakan standar internet termasuk nama dan domain user. Domain akan diganti dengan informasi yang lain sebagai keterhubungan langs-ung yang telah dibentuk atau terdapat mesin pembalas dalam path. SMTP menggunakan *domain name server* untuk semua alamat.Sekalipun *e-mail* digambarkan akan dapat sampai ke tempat tujuan seketika itu juga, kenyatannya tidak sesederhana itu. *E-mail Internet* tergantung juga pada teknologi "*store and forward*", yang berarti pesan-pesan akan ditahan dan disimpan dalam satu komputer atau lebih sepanjang perjalanan surat tersebut dan diteruskan lagi pada selang waktu yang tetap atau pada saat lalu lintas jaringan memungkinkan. Transmisi data SMTP menggunakan format yang sederhana. Semua teks pesan ditransfer ke dalam karakter ASCII 7 bit. Pada akhir suatu pesan akan dikenali menggunakan periode tunggal pada jalur tersebut. Jika dalam beberapa hal jalur pesan dimulsi dengan periode tertentu, beberapa saat kemudian ditambahkan oleh protokol untuk menghindari terjadinya hal yang membinggungkan dengan indikator akhir pesan.

F. Prosedur-prosedur pada SMTP

a) Mail

Ada 3 tahap pada transaksi mail SMTP :

1. MAIL command

Pada MAIL command ini akan diberikan identifikasi dari pengirim

MAIL <SP> FROM :<reverse-path> <CRLF>

Perintah ini memberitahukan kepada receiver SMTP bahwa ada transaksi email baru dan untuk mereset semua tabel state dan buffer, termasuk semua penerima atau mail data. Command ini akan memberikan *reverse-path* yang digunakan untuk melaporkan jika terjadi error. Jika diterima, maka receiver SMTP akan mengirimkan kode 250 OK *reply*

2. RCPT command

Disini akan diberikan informasi dari penerima

RCPT <SP> TO:<forward-path> <CRLF>Perintah ini akan memeberikan *forward path* yang akan mendefinisikan penerima. Jika diterima maka receiver SMTP akan mengirmkan kode 250 OK *reply* dan menyimpan *forward path* tersebut. Bila penerima tidak diketahui maka *receiver* SMTP akan mengirimkan kode 550 *Failure reply.* Prosedur kedua ini dapat diulangi sampai beberapa kali

3. Data *command*

Berisi data dari MAIL dan indikasi akhir dari transaksi

DATA <CRLF>

Jika diterima, maka *receiver* SMTP akan mengirimkan kode 354 Intermediate reply dan menganggap semua baris dari data adalah message text. Jika akhir dari teks diterima dan diterima dan disimpan. SMTP *receiver* akan mengirimkan 250 OK *reply* Akhir dari *e-mail* data ini juga berisi konfirmasi dari transaksi mail dan memberitahukan kepada receiver SMTP untuk memproses mail data dan mengirimkannya kepada pengirim.

Contoh:Mail from:<Smith@Alpha.ARPA>
R: 250 OK
S: RCPT TO:<Jones@Beta.ARPA>
R: 250 OK
S: RCPT TO:<Green@Beta.ARPA>
R: 550 No such user here
S: RCPT TO:<Brown@Beta.ARPA>
R: 250 O
S: DATA
R: 354 Start mail input; end with
<CRLF>.<CRLF>
S: Blah blah blah...
S: ...etc. etc. etc.

S: <CRLF>.<CRLF>
R: 250 OK

b) Forwarding

Ada 2 kasus *reply receiver* SMTP dari Forwarding ini, yaitu dimana informasi tujuan (*forward path*) salah, tetapi *receiver* SMTP mengetahui tujuan yang benar .ada 2 kategori perintah diantaranya:

a. 251 *User not local; will forward to <forward path> Reply* ini memberitahukan bahwa receiver SMTP mengetahui mailbox dari user ada pada host yang lain dan memperbaiki *forward path* yang sebelumnya untuk digunakan selanjutnya. Pada kasus ini receiver bertanggung jawab untuk mengirimkan pesan

b. 551 User not local; please try <*forward-path*>Perbedaan pada kasus pertama ialah bahwa receiver SMTP menolak menerima email untuk user ini dan pengirim harus me-*redirec*t email berdasarkan informasi yang disediakan oleh receiver SMTP atau mengirimkan pesan error pada user yang dituju.

Contoh :
S: RCPT TO:<Postel@USC-ISI.ARPA>
R: 251 User not local; will forward to <Postel@USC-ISIF.ARPA>
S: RCPT TO:<Paul@USC-ISIB.ARPA>
R: 551 User not local; please try <Mockapetris@USC-ISIF.ARPA>

c) Verifying and Expanding

Merupakan fitur tambahan dari SMTP, command ini digunakan untuk memverifikasi nama user atau untuk meng-*expand mailing list*. Digunakan dengan perintan VRFY dan EXPN, dengan argument karakter *string*. Untuk VRFY, argumennya adalah nama user dengan responnya adalah Nama lengkap dari user serta mailboxnya. Sedangkan untuk EXPN argumennya adalah *mailing list.*

Contoh :
VRFY
S : VRFY Smith
R : 250 Fred Smith <Smith@USC-ISIF.ARPA>
S : VRFY Smith
R : 251 User not local; will forward to <Smith@USC-ISIQ.Arpa>
S : VRFY Jones
R : 550 String does not match anything
S : VRFY Jones
R : 551 User not local; please try <Jones@USC-ISIQ.ARPA>
S : VRFY Gourzenkyinplatz
R : 553 User ambiguous
EXPN
S : EXPN Example-People
R : 250-Jon Postel <Postel@USC-ISIF.ARPA>
R : 250-Fred Fonebone <Fonebone@USC-ISIQ.ARPA>
R : 250-Sam Q. Smith <SQSmith@USC-ISIQ.ARPA>
R : 250-Quincy Smith <@USC-ISIF.ARPA: Q-Smith@ISI-VAXA.ARPA>
R : 250-<joe@foo-unix.ARPA>
R : 250 <xyz@bar-unix.ARPA>
S : EXPN Executive-Washroom-List
R : 550 Access Denied to You

d) Sending and Mailing

Sending adalah pengiriman keterminal dari suatu user, sementara mailing adalah pengiriman ke mailbox dari suatu user, karena fungsi keduanya hampir identik, maka fungsi keduanya digabungkan kedalam SMTP

Ada 3 perintah utama pada command sending, yaitu :

1. SEND <SP> FROM: <reverse-path> <CLRF>

Perintah ini berguna untuk mengirimkan mail data ke terminal dari user. Jika user sedang tidak aktif atau tidak mail tersebut tidak sampai, maka pada host akan muncul 450 Reply

2. SOML <SP> FROM: <reverse-path> <CRLF>

SOML ini merupakan sarana pengiriman email dimana tidak mengharuskan user tersebut aktif atau tidak, jika tidak aktif maka mail tersebut akan dimasukkan kedalam kotak masuk dari pengguna

3. SAML <SP> FROM: <reverse path> <CRLF>

Pada kasus SAML ini mail data yang datang pada suatu user dari suatu host akan langsung dimasukkan kedalam kotak masuk.

e) Opening and Closing

Saat kanal transmisi terbuka, maka ada sebuah pertukaran informasi yang menjamin bahwa host tersebut adalah host yang dimaksud *command* yang digunakan dalam membuat suatu koneksi pada kanal transmisi dan saat memutuskan koneksi adalah :

***HELO* <SP> <domain> <CRLF>** ***QUIT* <CRLF>**

Pada command HELO, host akan mengirimkan command yang mendefinisikan informasi

dirinya.

Contoh :
Memulai suatu koneksi
R: 220 BBN-UNIX.ARPA Simple Mail Transfer Service Ready
S: HELO USC-ISIF.ARPA
R: 250 BBN-UNIX.ARPA
Mengakhiri suatu koneksi
S: QUIT
R: 221 BBN-UNIX.ARPA Service closing transmission channel

f) Relaying

Forward-path biasanya adalah rute yang berupa bentuk "@x atau @y" dimana x dan y adalah nama host, bentuk ini akan membedakan alamat dan rute, dimana rute adalah cara bagaimana untuk mencapai alamat tersebut.Elemen dari forward-path akan dipindahkan menjadi *reverse-path* pada saat message dikirimkan dari satu *server* SMTP ke *server* SMTP yang lain.

Contoh
Pesan pemberitahuan bahwa e-*mail* yang dikirimkan tidak sampai
S: MAIL FROM:<>
R: 250 ok
S:RCPT
TO:<@HOSTX.ARPA:JOE@HOSTW.ARPA>
R: 250 ok
S: DATA
R: 354 send the mail data, end with .
S: Date: 23 Oct 81 11:22:33
S: From: SMTP@HOSTY.ARPA
S: To: JOE@HOSTW.ARPA
S: Subject: Mail System Problem
S:
S:Sorry JOE, your message to SAM@HOSTZ.ARPA lost.
S: HOSTZ.ARPA said this:
S: "550 No Such User"
S: .
R: 250 ok

g) Domains

Penggunaan nama domain menggantikan alamat dari nama host menjadi alamat global. Nama host digantikan dengan bentuk nama domain yang diikuti dengan nama host dan dipisahkan oleh tanda titik "."
Contoh
"USC-ISIF.ARPA", "Fred.Cambridge.UK" "PC7.LCS.MIT.ARPA"
adalah identifikasi nama host-dan-domain

h) Changing Roles

Command TURN digunakan untuk membalik peran dari 2 program yang berkomunikasi pada kanal transmisi.Jika program A adalah sender SMTP dan ia mengirimkan command TURN dan menerima ok reply (250) maka program A akan menjadi *receiver* dari SMTP. Jika kita menolak perubahan tersebut makan receiver akan mengirimkan 502 *reply*

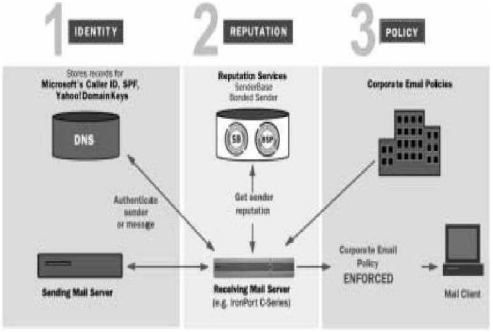
G. Penanganan SPAM dengan Metode SMTPi

Menurut Chris Brenton dan Cameron Hunt(2005:123) *Simple Mail Transfer Protocol* (SMTP) adalah digunakan untuk melakukan transfer *pesan e-mail* diantara sistem.SMTP menggunakan sebuah jenis koneksi yang bersifat *message-switched* atau aktif jika ada pesan yang masuk. Setiap pesan *e-mail* diproses sampai selesai sebelum session antara dua system diakhiri. Jika ada lebih dari satu pesan *e-mail* dikirimkan,maka sebuah sesi terpisah akan dibuat untuk setiap pesan *e-mail.* Konsep ini mengatakan bahwa SMTP menggunakan informasi yang ditulis pada amplop surat (*message header*) dan tidak melihat isi surat (*message body*). Daftar tujuan-tujuan e-mail diperoleh *user agent* dari *message header*. Dalam beberapa kasus, alamat tujuan biasanya sudah tercantum dalam *message header*. Dalam kasus lain, *user agent* mungkin perlu memperluas nama-nama dalam *mailing list*, membuang duplikat, dan mengganti nama-nama *mnemonic* dengan nama-nama yang aktual. Jika terdapat *blind carbon copies* (BCC), *user agent* perlu menyiapkan pesan sesuai dengan kebutuhan ini. Ide dasarnya adalah format-format dan gaya-gaya ganda yang dibuat manusia dalam antar muka (*interface*) pengguna yang digantikan oleh daftar standar untuk SMTP *send program*. SMTP *sender* mengambil pesan dari antrian dan mentransmisikan ke *host* tujuan lewat transaksi SMTP dengan membuat sebuah atau lebih koneksi TCP *port* 25 dari *host target.* Sebuah *host* mungkin saja mempunyai banyak SMTP *sender* yang aktif secara simultan jika ia mempunyai banyak *mail* yang harus dikirimkan dan ia juga mempunyai kapabilitas untuk menciptakan SMTP *receivers* jika ada permintaan.Ketika SMTP *sender* telah selesai megirimkan suatu pesan, maka ia akan menghapus alamat tujuan yang dimaksud dari daftar tujuan. Ketika semua alamat tujuan untuk pesan telah selesai diproses, pesan tersebut

dihapus dari antrian. Dalam pemrosesan antrian, SMTP *sender* dapat melakukan bermacam-macam optimasi. Jika sebuah pesan dikirimkan untuk beberapa pengguna pada satu buah *host*, pesan informasi tersebut hanya perlu dikirimkan satu kali. Jika pesan-pesan tersebut sudah siap untuk dikirimkan ke sebuah *host*, maka SMTP *sender* dapat membuka sebuah koneksi TCP, mentransfer semua pesan tersebut, kemudian menutup koneksi, daripada membuka dan menutup sebuah koneksi berulang-ulang untuk tiap-tiap pesan.SMTP *sender* harus berurusan dengan bermacam-macam kesalahan. *Host* tujuan bisa saja tidak dapat dijangkau, tidak beroperasi, atau koneksi TCP bisa gagal ketika surat sedang dalam proses transfer. Pengirim bisa memasukkan kembali surat tersebut ke dalam antrian untuk dikirimkan beberapa waktu kemudian, tetapi bila setelah periode waktu tertentu surat masih tidak dapat dikirimkan, maka surat tersebut tidak jadi dikirimkan agar tidak memenuhi antrian untuk jangka waktu yang tidak terbatas. Kesalahan yang paling umum adalah kesalahan menuliskan alamat tujuan yang dapat terjadi pada saat memasukkan *input* atau karena alamat tujuan memiliki alamat baru di *host* yang berbeda. SMTP *sender* harus mengalihkan (*redirect*) pesan jika memungkinkan atau mengirimkan pesan kesalahan pada pengirim.Menurut Anjik Sukmaji dan Rianto(2008:143) SMTP adalah *protocol* standar yang dapat dipergunakan untuk mengirim *e-mail* ke suatu tujuan yang sudah ditentukan..SMTP berjalan di *text based protocol*.Sehingga semua yang dikirimkan dalam bentuk ASCII.Tujuan utama dari penggunaan SMTP adalah untuk *transfer mail reliably and efficiently*.proses SMTP akan dimulai
dengan insial jabat tangan(*handshaking*) selesai saluran transmisi akan terbentuk dan
SMTP akan diberikan respon yang menandakan respond yang menandakan *command* tersebut telah diterima dan terjadi *error condition*. SMTP client juga disebut sebagai *Mail User Agent* (MUA).Sedangkan SMTP *server* lebih dikenal dengan dengan *Mail Transfer Agent* (MTA). Untuk *attachment* yang berbentuk biner akan dilakukan proses *encode* dalam bentuk 8 bit MIME(*Multipurpose Internet Mail Extension*).SMTP menggunakan *port* TCP 25.SMTP bersifat "*push"protocol* dimana hanya bersifat mengirimkan untuk "*pull*" protocol dari remote server(*mail server)* digunakan*l Post Office Protocol*(POP3). atau *internet mail Acces Protocol*(IMAP). Memastikan akuntabilitas dari pengirim dalam pengiriman *e-mail* yang aman dan dapat dipercaya sangat diperlukan. SMPTPi adalah next-generation dari infrastruktur e-mail. SMTPi ini memiliki 3 f*ramework* dengan komponen penting yaitu identitas, reputasi dan policy tertentu dari suatu *system messaging* yang baru dan aman yang dibuat diatas SMTP. "i" dibelakang SMTP itu adalah untuk *identity*. Perpindahan dari identitas dan reputasi dari *mail* sistem akan membuat pengirim menjadi mudah dipantau dan mengurangi masalah-masalah yang ada dengan *e-mail*. SMPTi ini berpedoman pada skema autentifikasi, reputasi pengiriman data dan kemampuan untuk membuat sebuah kebijakan *e-mail* berdasarkan atas kedua aspek diatas.SMTP dapat digunakan oleh pengguna sebagai protokol untuk saling menukar *e-mail* antar pengguna dalam komputer yang sama atau komputer yang lain.layanan SMTP yaitu:

1. Untuk mengirimkan satu pesan kesatu atau banyak penerima
2. untuk mengirim pesan termasuk teks,suara,video dan grafik.
3. untuk mengirim pesan ke user dalam satu jaringan ataupun diluar jaringan

H. Implementasi SMPTi



Gambar II.1 Implementasi SMTPi(Iron Port 2004)

Menurut Iron Por(2004) Implementasi SMTP

terbagi menjadi

1. Identity

Secara akurat akan menunjukkan identitas dari pengirim sehingga membuat penerima *e-mail* yakin bagaimana ia harus memperlakukan message yang masuk, terkait dengan reputasi dari pengirim. Dengan melakukan ini, makan akan lebih mudah untuk menghapus *spam* dari *inbox* penerima Membuat suatu mekanisme

identitas yang menyeluruh untuk *e-mail* adalah masalah yang tidak mudah dan perlu dilakukan secara bertahap.

Mengetahui identitas pada pengiriman *e-mail* memiliki 2 keuntungan :Memungkinkan user untuk mengetahui reputasi dari pengirim dan dapat membuat keputusan yang baik bagaimana harus memperlakukan *e-mail* dari pengirim tersebut berdasarkan reputasinya (diterima atau ditolak) Memberikan informasi bagi pengirim untuk memperbaiki perilaku mereka untuk meningkatkan reputasi mereka dalam pengiriman *e-mai.*

a. Server Level Identity *berdasarkan IP address*

Identitas dari mekanisme level server adalah alamat IP dari pengirim. Sebuah alamat IP dapat diverifikasi dan dimanage, dan hampir tidak mungkin untuk dipalsukan, karena dikirimkan dengan menggunakan koneksi TCP/IP. Apabila alamat IP salah dalam pengartian *make two-way* SMTP *conversation* tidak mungkin terjadi karena pengembalian paket untuk melakukan komunikasi SMTP tidak bisa dirutekan kembali ke alamat IP asal.IP address sendiri memiliki keterbatasan sebagai identitas, dan keterbasan ini akan memungkinkan untuk dikembangkannya autentifikasi level Domain

b. Domain Level Identity

Sistem Autentifikasi tingkat domain akan memungkinkan perusahaan untuk menentukan mail server mana yang diperbolehkan untuk mengirimkan *e-mail* dengan menggunakan nama doamin. Hal ini dapat mencegah bentuk-bentuk pemalsuan alamat *e-mail* seperti "pishing" dan "j*oe-jobs*". "*Pishing*" terjadi jika pengirim menggunakan nama doamin yang sudah banyak dikenal seperti *e-bay.com* atau *yahoo.com* pada "*FROM*: " *address.* "Joe-*jobs"* merupakan bentuk lain pemalsuan yang berbahaya, dimana pengirim membuang e-mail yang tidak diinginkan dan memalsukan "*envelope*" addrss dari orang yang tidak terkait sama sekali. Maka hal ini akan membuat pengirim terhindar dari pengiriman kembali message yang tidak diinginkan dan mendiskriditkan orang yang alamatnya telah dipalsukan tersebut.Untuk beberapa tahun kedepan, identitas dari level-domain dapat dilakukan dengan menggunakan standar seperti *Sender Policy Framework* (SPF), *Calle*r-ID dan Domainkeys, tetapi untuk tiap jenisnya ada keuntungan dan kerugiannya.SPF dan *Caller-ID* berusaha untuk mengatasi keterbatasan pembuatan list "*authorized source IP'*s" pada "*envelope sender"* atau *header* dari domain pada DNS. Sedangkan *Domain Keys*, menggunakan *private key* untuk enkripsi pada setiap pengiriman message.Solusi yang paling baik, belum dapat ditentukan, penggunaaan dari *header cryptography* yang akan memungkinkan user mengidentifikasikan diri mereka sendiri dalam berbagai tingkatan (individu, organisasi dan perusahaan).SPF merupakan solusi dengan biaya rendah untuk masalah "*unwanted bounce"* tetapi tidak mengatasi spam untuk masalah imitasi alamat *e-mail* dan membatasi beberapa aplikasi *e-mail* seperti pengiriman. Sementara *Caller ID* mengatasi masalah imitasi alamat *e-mail* tetapi biaya lebih tinggi untuk implementasi pengiriman *e-mail.* Lalu *Domain Keys* merupakan solusi yang paling baik untuk pemalsuan, tetapi sangat rumit, membutuhkan perubahan dari infrastruktur e-mail yang sudah ada sekarang

c. Universal Identity

Masalah utama dari autentifikasi tingkat Domain adalah bahwa ia tidak dapat mengidentifikasikan user individual. "*Universal Identity*" akan menyediakan bentuk identitas yang lebih fleksibel dan aman dengan menggunakan digital certificates. Model ini memungkinkan pengirim untuk mengidentifikasikan diri mereka sendiri pada berbagai tingkatan seperti individual, organisasi atau suatu perusahaan.

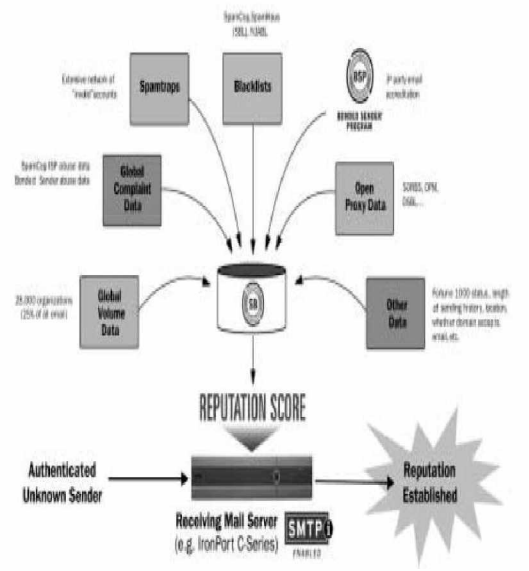
2. Reputation

Reputasi dari pengirim dapat ditelusuri dengan memonitor *history mail*-nya. *Track service* dari reputasi pengirim merupakan range dari parametere terukur, seperti volume dari *e- mail* yang dikirimkan, bertentangan, Negara asal, kehadiran dari *relay open proxy*, konfigurasi DNS yang terkait, serta data-data terkait lainnya. Paramater ini dapat menentukan reputasi dari pengirim.Tidak seperti *blocklist*, yang merupakan pengaruh dari generasi pertama dari servis untuk menentukan reputasi pengirim, servis reputasi generasi kedua seperti "*Sender Base*" akan menyediakan data secara detil (nilai reputasi dari -10 hingga +10) yang akan memungkinakn penerima menentukan kebijakannya sendiri dan thresholds. *SenderBase* adalah *Open Service*, dimana *system administrator* dan *open source spam filter* dapat mengakses tanpa mengeluarkan biaya. Untuk dapat berguna, reputasi dari *e-mail* dan pelayanan akreditasi harus terbuka, memiliki data-data yang otentik dan Seiring dengan berkembangnya SMTPi, penerima dari *e-mail* akan menerpakan batasan yang lebih dari *e-mail* yang berasal dari sumber yang tidak memiliki

identitas dan reputasi. Perpindahan SMTP menuju sistem SMTPi (*identity and reputation)* akan membuat *e-mail l*ebih aman dan terjamin

3. Sender Base

Merupakan bentuk *database* pertama dari pengirim *e-mail.* Memiliki *host* pada *www.senderbase.org*, *SenderBase* merupakan semacam badan pelaporan servis dari *e-mail,* menyediakan data-data dari ISP atau perusahaan yang nantinya akan digunakan untuk menentukukan keputusan terhadap suatu sumber *e-mail*.Diguanakan oleh lebih dari 30.000 *administrator mail*, *SenderBase* memonitor jumlah (volume) pengiriman *e-mail,* tingkat keluham, *account* dari "*spamtraps*", daerah asal dari *mai*l, informasi yang dibatasi, status dari *open proxy* serta parameter-parameter lain yang digunakan untuk menentu kualitas dari pengirim. Semuanya ini akan sangat berguna bagi *administrator*



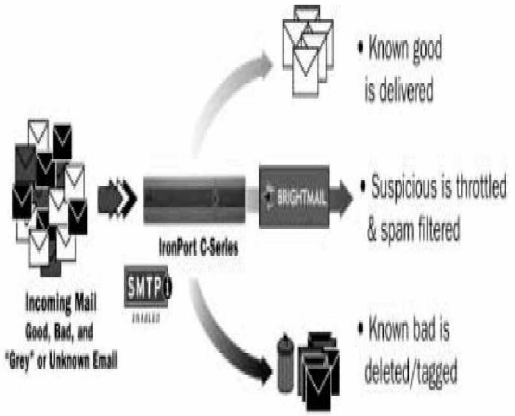
Gambar II.2 Filterisasi SPAM dengan Metode SMTP i(Iron Port 2004)

4. Bonded Sender Program

Merupakan servis akrediatai *e-mail. Bonded Sender* menggunakan mekanisme pasar yang unik untuk memastikan bahwa *e-mail* komersial yang terpercaya tidak dikatergorikan kedalam *spam* dan tidak terkena *filter spam*. Pengiri dari *e-mail* komersial yang tidak termasuk dalam spam, harus mendaftarkan *e-mail* nya dengan membayar sujumah uang. Jika end user memberikan keluhan bawah pesan yang masuk adalah bukan yang diinginkan, maka dapat dilakukan *claim*. Mekanisme ini menjamain bahwa hanya perngirim yang terpercaya yang dapat menggunakan program ini.

5. Policy

Setelah mengautentifikasi *e-mail* dari pengirim dan mengetahui reputasinya, penerima *e-mail* perlu untuk membuat suatu cara yang sesuai untuk membuat suatu kebijakan *e-mail* berdasarkan informasi yang sudah diperoleh sebelumnya Sekarang ini kebanyakan *e-mail gateway* memproses semua *e-mail* yang masuk melalui suatu penyaringan *spam*. Metode ini akan meningkatkan biaya dari infrastruktur dan mengurangi tingkat efektifitas dari pendeteksian *spam*.Sebuah solusi kebijakan *e-mail,* respon yang mendukung variabel yang berdasarkan atas kualiatas dan tingkat kepercayaan atas sumber dari *e-mail* yang masuk. *E-mail* yang berasal dari pengirim yang diketahui dapat dilalukan dari Penyaringan spam , sedangkan *e-mail* yang berasal dari sumber yang tidak dipercaya dapat dihapus dan *e-mail* yang berasal dari sumber yang mencurigakan dapan diblok dan dilewatkan melalaui penyaringan *spam* yang sangat sensitive



Gambar II.3 Penyortiran e-mail masuk(Iron Port 2004)

\

III. PENUTUP

3.1. Kesimpulan

Spam adalah pengiriman sejumlah *message* yang sama yang memaksa orang lain yang tidak menginginkan adanya *message* yang mau tidak mau harus menerimanya. Tidak semua mail yang berisi propaganda dapat di-kategorikan sebagai *Spam*. Jika isi *e-mail* tersebut relevan dengan minat penerima *e-mail,* maka *e-mail* tersebut bukan *Spam* bagi pengguna *e-mail* tersebut. Hal

ini yang menyebabkan usaha untuk mengirimkan Spam terus dilancarkan, karena penerima yang tepat akan benar-benar membalas atau berminat terhadap isi yang ditawarkan.SMTPi merupakan salah satu solusi untuk penanganan *spam* yang berbasiskan SMTP. Kelebihan dari SMPTi ini adalah karena adanya 3 aspek utama yaitu Identitas,Reputasi dan Kebijakan. Walaupun demikian,bisa dikatakan tidak mudah menghindari *spam* karena teknik yang digunakan hampir menyerupai pengiriman *e-mail* normal. Kebanyakan *spam* yang ada di *internet* adalah berupa iklan, promosi dari suatu produk. *Spam* ini sangat menguntungkan bagi pengirim, karena hanya membutuhkan biaya yang kecil, sementara tidak sebaliknya jika kita memandang dari sisi penerima. *Spam* adalah tindakan yang tak bertanggung jawab. *Spam* jelas-jelas merugikan banyak pihak, sementara hanya menguntungkan satu dua pihak. *Spam* pun tak diinginkan praktis oleh semua orang. Jadi, demi masa depan yang baik, adalah seharusnya spam berkurang atau ditiadakan sama sekali. Jikalau nanti Indonesia sudah menyusul dan mulai membuat peraturan seputar *cyberspace* termasuk untuk mengatur *spamming*,

marilah kita semua bersama-sama mendukungnya. Atau kalau belum, marilah mulai mendorong pihak-pihak yang di atas sana untuk segera merealisasikan hal ini.

3.2. Saran

Spam dengan menggunakan SMTPi Sangat perlu, karena Memastikan akuntabilitas dari pengirim dalam pengiriman e-mail yang aman dan dapat dipercaya sangat diperlukan. SMPTPi adalah generasi terbaru dari infrastruktur *e-mail*. SMTPi ini memiliki 3 *framework* dengan komponen penting – identitas, reputasi dan policy tertentu dari suatu system messaging yang baru dan aman yang dibuat diatas SMTP. Perpindahan dari identitas dan reputasi dari mail system akan membuat pengirim menjadi mudah dipantau dan mengurangi masalah-masalah yang ada dengan *e-mail*. SMPTi ini berpedoman pada skema *autentifikasi*, reputasi pengiriman data dan kemampuan untuk membuat sebuah keamanan *e-mail* berdasarkan atas kedua aspek diatas.

DAFTAR PUSTAKA

Bagnall Brian.2000.*E-mail Virus Protection* Syngress Publishing, Inc.800 Hingham StreetRockland, MA 02370

Brenton, Chris dan Cameron Hunt.2005.,*Network Security*.Jakarta: Elexmedia Komputindo.

Costales Bryan , Marcia Flynt.2005. sendmail Milters A Guide for Fighting Spam Addison Wesley Professional Pearson Education, Inc. rights and Contracts Department. Lake Street Upper Saddle River, NJ 07458

Raharjo, Budi.2005.Keamanan Sistem Informasi Berbasis Internet. Jakarta :PT Indocisc dan Bandung:PT Insan Infonesia

Sukmaji Anjik dan Rianto.2008.Jaringan Komputer.Yogyakarta:Andi

Zdziarski Jonathan A..2005. Ending spam : Bayesian content filtering and the art of statistical language classification. No Starch Press, Inc. 555 De Haro Street, Suite 250, San Francisco

Referensi Web

Iron Port. 2004.*An e-mail security architecture* IronPort Systems, Inc.1100 Grundy Lane, Suite 100 San Bruno, California 94066 (http://www.ironport.com/pdf/ironport_e-mail_authentication_wp.pdf) (diakses 20 Agustus 2009)

Krawetz Neal.2004.*Anti-Spam Solutions and Security*. (http://www.securityfocus.com/infocus/1763) (Diakses 20 Agustus 2009)

B.Postel Jonathan .1982.Simple mail transfer protocol Information Sciences Institute University of Southern California 4676 Admiralty Way Marina del Rey, California90291(http://www.freesoft.org/CIE/RFC/821/2.htm) (Diakses 20 Agustus 2009)

KAJIAN PENERAPAN SISTEM INFORMASI KARYAWAN BERBASIS *WEB* BERDASARKAN PENDEKATAN TAM

Mochamad Wahyudi

Program Pascasarjana Magister Ilmu Komputer STMIK Nusa Mandiri
Jl. Salemba Raya No. 5 Jakarta Pusat (10250) Indonesia
wahyudi@nusamandiri.ac.id

Abstract

The purposes of this study are to find the dominant factors which correlate and influence the level of technology acceptance, particularly Web-Based Employee Information System to its end users, of which are the employees of Bina sarana Informatika (BSI), and to know how an accepted model of new technology in a form of Web-Based Employee Information System is emplemented in a tertiary education institution.

The Web-Based Employee Information System used in this study is one of information systems implemented by BSI, all of which were web-based designed that require an internet browser, such as: Internet Explorer or Mozilla Firefox. It is computer-accessible to employees both from within or outside BSI campus since the system is connected with local computer network via both intranet as well as internet facilities. This Web-based Employee Information System, which is stored in a web-server of the Internet, has a domain address "bsi.ac.id" and it is accessible via BSI website at http://www.bsi.ac.id. Some facilities available are: employee data processing, academic service, prospective students information, courses and programs, Jabatan Fungsional Dosen*.*

The instrument for this study is questioner with Semantic Differential Scale ranging from 1 to 7, representing answers ranging from 'extremely disagree' to 'extremely agree'. This study also makes use of both Technology Accepted Model (TAM) to describe the relationship between factors influencing the use of Web-Base Employee Information System and Structural Equation Modelling (SEM) to analyse data. The software used are AMOS and SPSS for Mindows 16.0.1 version.

The result of this study is useful for identifying and putting in mind the role of Web-Based Employee Information System, which is accessible at http://www.bsi.ac.id, as a facility to support employees' working accomplishment.

Key words : Web-Based Employee Information System, Technology Accepted Model (TAM), Structural Equations Modelling (SEM), Analysis of MOment Structure (AMOS)

I. PENDAHULUAN

1. Latar Belakang

Penelitian ini dilakukan kampus BSI, dimana kampus BSI menerapkan sistem *paperless* untuk semua layanan, baik terhadap mahasiswa maupun karyawan. Sistem tersebut didukung dengan dibangunnya aplikasi-aplikasi komputer, baik berupa aplikasi *desktop* maupun aplikasi berbasis *web (Web Base Applications)* yang keseluruhannya dibangun sendiri *(Taylor Made System)* oleh BSI melalui salah satu unit kerja yang bernama Biro Teknologi Informasi Bina Sarana Informatika (BTI BSI).

Penulis ingin meneliti salah satu sistem yang telah dibangun pada kampus BSI. Objek penelitian yang akan penulis teliti adalah Sistem Informasi Karyawan Berbasis *Web* yang dapat diakses menggunakan *browser internet* pada alamat http://www.bsi.ac.id.

Penelitian ini menganalisa faktor-faktor apa saja yang mempengaruhi penggunaan Sistem Informasi Karyawan Berbasis *Web*. Faktor-faktor yang diteliti meliputi adanya persepsi kemudahan menggunakan *(Perceived Ease of Use)*, adanya manfaat *(Perceived Usefulness)*, adanya niat untuk menggunakan *(Intention to Use)* dan penggunaan *Website (Website Usage)*.

2. Tujuan dan Manfaat Penelitian

Tujuan yang ingin penulis capai pada penelitian ini adalah mengidentifikasi faktor-faktor dominan apa saja yang dapat mempengaruhi karyawan dalam menggunakan Sistem Informasi Karyawan Berbasis *Web* dan

bagaimana model penerimaan penggunaan Sistem Informasi Karyawan Berbasis *Web* pada BSI.

Manfaat dari penelitian ini diharapkan dapat membantu untuk mengidentifikasi dan mengingatkan peran penggunaan Sistem Informasi Karyawan Berbasis *Web* yang dapat diakses oleh seluruh karyawan BSI melalui alamat http://www.bsi.ac.id, sebagai sarana pendukung untuk dalam menyelesaikan pekerjaan dan masukan kepada pihak manajemen BSI dalam rangka untuk peningkatan pelayanan dan pengembangan Sistem Informasi Karyawan Berbasis *Web*.

II. PEMBAHASAN

Tinjauan Pustaka

1. Sistem dan Informasi

Sistem adalah sekelompok elemen yang terintegrasi dengan maksud yang sama untuk mencapai suatu tujuan ((McLeod 2004), 13).

Jerry Fitzgerald mendefinisikan sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau menyelesaikan suatu sasaran tertentu (Jogiyanto 2006).

Informasi dapat didefinisikan sebagai data yang telah diproses atau data yang memiliki arti (McLeod 2004), atau data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya (Jogiyanto 2006).

2. Sistem Informasi

Henry C. Lucas mendefinisikan sistem informasi adalah suatu kegiatan dari prosedur-prosedur yang diorganisasikan, bilamana dieksekusi akan menyediakan informasi untuk mendukung pengambilan keputusan dan pengendalian didalam organisasi (Jogiyanto 2006).

Robert A. Leitch mendefinisikan sistem informasi adalah suatu sistem didalam organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial, dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan (Jogiyanto 2006).

Sistem informasi adalah sebuah sistem yang mengarah pada penggunaan teknologi komputer dalam organisasi yang menyajikan informasi kepada pemakai.

3. *Internet*

Secara tradisional *internet* memiliki empat aplikasi utama (Tanenbaum 2000), yaitu : *electonic Mail (e-mail), News, Remote Login* dan *Transfer File*. Sampai awal tahun 1990-an, *internet* banyak dipakai oleh para akademisi, pemerintah, para peneliti industri. Sebuah aplikasi yang disebut *World Wibe Web* (WWW) mengubah semua itu dan membantu jutaan pengguna baru, non akademisi ke jaringan. Aplikasi ini ditemukan oleh fisikawan CERN Tim Berners-Lee, tanpa mengubah fasilitas-fasilitas yang telah ada namun membuatnya lebih mudah digunakan.

4. Model Persamaan Struktural

Model Persamaan Struktural atau biasa disebut dengan *Structural Equation Modelling (SEM)* adalah sekumpulan teknik-teknik analisis statistika yang mengkombinasikan beberapa aspek yang terdapat pada analisis jalur dan analisis faktor konfirmatori untuk mengestimasi beberapa persamaan secara simultan dan berjenjang. Hubungan simultan dan berjenjang yang dimaksud dibangun antara satu atau beberapa variabel dependen dengan satu atau beberapa variabel dependen. Masing-masing variabel dependen dan independen dapat berbentuk faktor atau konstruk yang dibangun dari beberapa variabel indikator.

SEM merupakan gabungan dari dua metode statistik yang terpisah, yaitu analisis faktor *(Factor Analyst)* yang dikembangkan pada bidang psikologi atau psikometri serta model persamaan simultan *(Simultaneus Equation Modelling)* yang dikembangkan pada bidang ekonometrika (Ghozali 2005). SEM juga merupakan teknik statistik yang mampu menganalisis variabel laten, variabel indikator, dan kesalahan pengukuran secara langsung. SEM ini juga memiiki keunggulan dibandingkan dengan metode statistik multivarensi *(Multivariate Statistic)* yang lain, karena dalam variabel laten dimasukan kesalahan pengukuran dalam model.

Menurut Hair (1998) tahapan pemodelan dan SEM dibentuk dalam tujuh langkah (Ghozali 2005), yaitu : pengembangan model secara teori, pengembangan diagram jalur (*Path Diagram*, konversi diagram jalur *(Path Diagram)* kedalam persamaan, pemilihan jenis *input* matriks dan estimasi model yang diusulkan, penilaian identifikasi

model struktural, penilaian kriteria *Goodness of fit* dan interpretasi dan modifikasi model.

5. Model Penerimaan Teknologi

Model Penerimaan Teknologi atau biasa disebut dengan *Technology Accepted Model* (TAM) digunakan untuk memprediksi penerimaan pengguna terhadap pengggunaan teknologi baru. Model yang dikenalkan oleh Fred D. Davis pada tahun 1989 ini merupakan model yang paling banyak dipergunakan dalam penelitian sistem informasi, karena menghasilkan validitas yang baik.

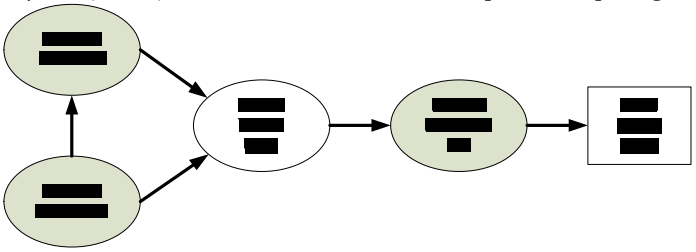
TAM merupakan adaptasi dari teori yang dikembangkan oleh Fishbein, yaitu *Theory of Reasoned Action* (TRA) yang merupakan teori tindakan yang berlandaskan dengan satu asumsi bahwa reaksi dan persepsi seseorang terhadap sesuatu hal akan menentukan sikap dan perilaku orang tersebut. Reaksi dan persepsi pengguna teknologi informasi akan mempengaruhi sikapnya dalam penerimaan teknologi informasi, yaitu salah satu faktor yang dapat mempengaruhi adalah persepsi pengguna antar kemanfaatan dan kemudahan penggunaan teknologi informasi sebagai suatu tindakan yang beralasan dalam kontek penggunaan teknologi informasi sehingga alasan seseorang dalam melihat manfaat dan kemudahan penggunaan teknologi informasi menjadi tindakan orang tersebut dapat menerima penggunaan teknologi informasi.

Model dasar dari pembentukan sikap yang mempengaruhi perilaku seseorang, berdasarkan TAM menggambarkan hubungan antara (Davis 1989) :

1. *Perceived Ease of Use (PEoU)*
   Menyatakan tingkat kepercayaan bahwa teknologi baru akan mudah untuk dipakai dan terbebas dari usaha.
2. *Perceived Usefulness (PU)*
   Menyatakan tingkat kepercayaan bahwa penggunaan teknologi baru akan meningkatkan pencapaian.
3. *Attitude Toward Using (ATU)*
   Menyatakan sikap pengguna *(user)* ke arah menggunakan teknologi baru.
4. *Behavioral Intention to Use (ITU)*
   Menyatakan perilaku pengguna *(user)* ke arah berlanjutnya penggunaan sebuah teknologi baru yang dianggap memberikan manfaat.
5. *Actual System Usage (ASU)*
   Menyatakan pengguna *(user)* benar-benar menggunakan teknologi baru secara nyata karena merasakan manfaatnya.

Tinjauan Studi

Penelitian pertama berkaitan dengan penggunaan SEM dan TAM dilakukan oleh Fred D. Davis yang membahas mengenai *"Perceived Usefulness, Perceived Ease of Use and User Acceptance of Information Technology"*. Penelitian tersebut dilakukan untuk menguji variabel-variabel yang dapat memprediksi tingkat penerimaan pengguna komputer terhadap pengguna. Penelitian tersebut menunjukan bahwa *Perceived Usefulness* dan *Perceived Ease of Use* merupakan penentu dasar dari penggunaan komputer, selain itu penggunaan teknologi *(usage)* dipengaruhi oleh tingkatan penerimaan terhadap teknologi. Model penerimaan teknologi yang dikembangkan oleh Fred D. Davis dapat dilihat pada gambar berikut.

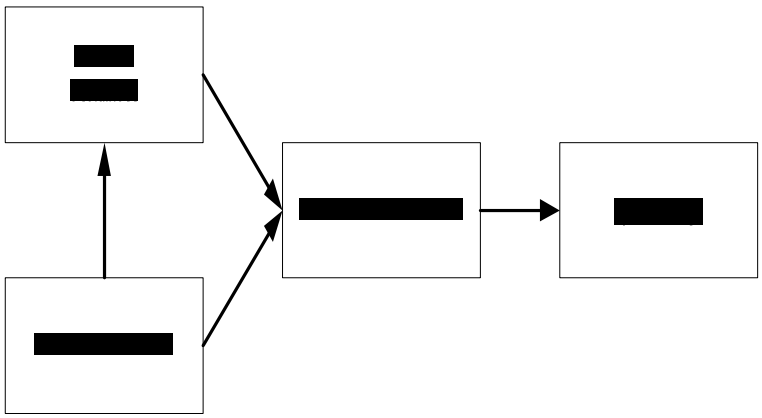


Gambar 1. *Technology Accepted Model (*Davis 1989)

Telah dilakukan beberapa penelitian yang berkaitan dengan penerimaan *Knowledge Management System* dalam suatu perusahaan terhadap perilaku penggunanya (karyawan), diantaranya penelitian yang dilakukan oleh Money dan Turner pada tahun 2004 dengan

judul penelitiannya "*Applications of Technology Acceptance Model to a Knowledge Management System"*. Penelitian tersebut dilakukan untuk menguji variabel-variabel yang dapat memprediksi tingkat penerimaan *Knowledge Management System* terhadap pengguna.

Penelitian ini menunjukan bahwa *Perceived Usefulness* dan *Perceived Ease of Use* merupakan penentu dasar penggunaan *Knowledge Management System.* Pada gambar berikut akan diperlihatkan model yang dikembangkan oleh Money dan Turner pada tahun 2004.

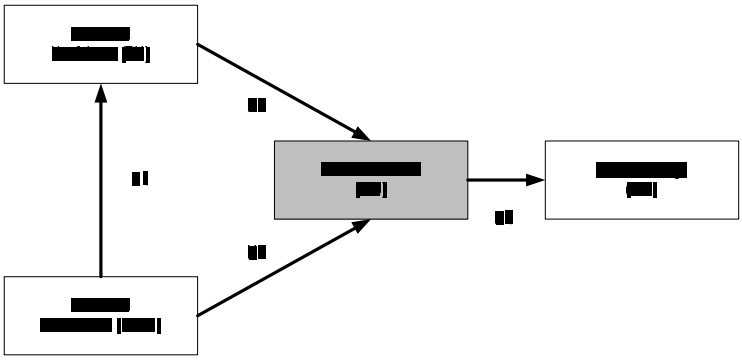


Gambar 2. *Technology Accepted Model* (Money 2004)

Kerangka Konsep

Penelitian ini merupakan salah satu penelitian lanjutan yang dikembangkan dari teori TAM yang diperkenalkan oleh Fred D. Davis pada tahun 1989, yang menguji dua faktor penerimaan teknologi yaitu *Perceived Usefulness (PU)* dan *Perceived Ease of Use (PEoU).* Pada penelitian ini diajukan konsep TAM berdasarkan Money dan Turner pada tahun 2004 (Money 2004).

Penelitian ini menggambarkan penggunaan TAM yang mensyaratkan bahwa karyawan menggunakan Sistem Informasi Karyawan Berbasis *Web* dipengaruhi oleh faktor-faktor seperti terlihat pada gambar berikut :



Gambar 3. Model penelitian "Kajian Penerapan Sistem Informasi Karyawan Berbasis *Web* Berdasarkan Pendekatan TAM"

Sedangkan indikator-indikator yang digunakan untuk mengukur masing-masing variabel laten dapat dilihat pada Tabel 1.

Tabel 1. Indikator dari faktor yang mempengaruhi karyawan dalam penggunaan Sistem Informasi Karyawan Berbasis *Web*

No	Variabel Laten	Indikator
1	*Perceived Ease of Use* (PEoU)	1. Kemudahan untuk mengakses 2. Kemudahan untuk dipelajari/dipahami 3. Kemudahan untuk digunakan 4. Kemudahan untuk berinteraksi
2	*Perceived Usefulness* (PU)	1. Mempertinggi efektifitas 2. Menjawab kebutuhan informasi 3. Meningkatkan kinerja 4. Meningkatkan efisiensi
3	*Intention to Use* (ITU)	1. Penambahan *software* pendukung 2. Motivasi tetap menggunakan 3. Memotivasi ke pengguna lain
4	*Website Usage* (WU)	1. Memahami cara penggunaan 2. Menyampaikan kepuasan 3. Frekuensi penggunaan

Desain Penelitian/Metodologi

Penelitian ini merupakan penelitian mengenai hubungan sebab akibat (kausal) dari variabel-variabel yang akan diteliti, sehingga dari penelitian ini diharapkan dapat diidentifikasi bagaimana dan apa saja faktor dominan yang mempengaruhi karyawan dalam penggunaan Sistem Informasi Karyawan Berbasis *Web* sebagai sarana pendukung dalam menyelesaikan pekerjaan di Kampus BSI.

Populasi diambil dari karyawan pada BSI yang menggunakan Sistem Informasi Karyawan untuk menunjang pekerjaannya yang berjumlah 1.073 karyawan. Dalam penelitian yang menggunakan SEM, besarnya ukuran sampel sangat berpengaruh. Hal ini disebabkan karena ukuran sampel memberikan dasar untuk mengestimasi *Sampling Error*. Dengan estimasi menggunakan *Maximum Likelihood* (ML), jumlah sampel minimal yang diperlukan 100 atau antara 100 sampai 200 sampel. Hal ini disebabkan karena apabila lebih besar atau sangat besar, maka akan menghasilkan perbedaan yang signifikan, sehingga *Goodness of Fit* menjadi tidak bagus (jelek). Hal serupa juga akan terjadi apabila jumlah sampel kurang dari 100.

Pada penelitian ini jumlah sampel yang penulis ambil adalah sebesar 110 sampel dari total populasi 1.073 karyawan yang menggunakan Sistem Informasi Karyawan Berbasis *Web*.

Metode pengumpulan data yang penulis lakukan pada penelitian ini adalah dengan cara menyebarkan kuesioner atau angket diberikan kepada karyawan BSI dengan teknik *Simple Random Sampling*.

Instrumentasi pada penelitian ini adalah berupa kuesioner atau angket yang menggunakan *Semantic Differential Scale* dengan range 1 sampai 7 untuk jawaban sangat tidak setuju sampai jawaban sangat setuju.

Kisi-kisi instrumen yang diperlukan untuk mengetahui faktor yang mempengaruhi karyawan BSI dalam menggunakan Sistem Informasi Karyawan Berbasis *Web* dapat dilihat pada Tabel 2.

Tabel 2. Kisi-kisi penelitian faktor yang mempengaruhi karyawan dalam penggunaan Sistem Informasi Karyawan Berbasis *Web*

No	Variabel Laten	Indikator	Jumlah Item	No. Item Instrumen
1	*Perceived Ease of Use* (PEoU)	X1.Kemudahan untuk mengakses	2	1 dan 2
		X2. Kemudahan untuk dipelajari/dipahami	2	3 dan 4
		X3. Kemudahan untuk digunakan	2	5 dan 6
		X4. Kemudahan untuk berinteraksi	2	7 dan 8
		Jumlah	8	
2	*Perceived*	Y1. Mempertinggi efektifitas	2	9 dan 10

	Usefulness (PU)	Y2. Menjawab kebutuhan informasi	2	11 dan 12
		Y3. Meningkatkan kinerja	2	13 dan 14
		Y4. Meningkatkan efisiensi	2	15 dan 16
		Jumlah	8	
3	*Intention to Use* (ITU)	Y5. Penambahan *software* pendukung	2	17 dan 18
		Y6. Motivasi tetap menggunakan	2	19 dan 20
		Y7. Memotivasi ke pengguna lain	2	21 dan 22
		Jumlah	6	
4	*Website Usage* (WU)	Y8. Memahami cara penggunaan	2	23 dan 24
		Y9. Menyampaikan kepuasan	2	25 dan 26
		Y10. Frekuensi penggunaan	2	27 dan 28
		Jumlah	6	
		Total	**28**	

5. Teknik Analisis Data

Teknik analisis data yang penulis lakukan pada penelitian ini adalah analisa statistik deskriptif dan analisa statistik inferensial.

Hasil Penelitian dan Pengujian

1. Hasil Penelitian

a. Analisis Statistik Deskriptif

Pengujian atau analisa terhadap statistik deskriptif yang memberikan penjelasan berupa nilai nilai rata-rata (mean), standar deviasi, varian, maksimum, range, kurtosis dan skewness dapat dilihat pada hasil pengolahan data kuesioner menggunakan *software SPSS for Windows* versi 16.0.1 seperti terlihat pada tabel 3.

Tabel 3. Hasil Pengujian Statistik Deskriptif

Descriptive Statistics

	N	Range	Minimum	Maximum	Mean	Std. Deviation	Variance	Skewness		Kurtosis	
	Statistic	Statistic	Statistic	Statistic	Statistic	Statistic	Statistic	Statistic	Std. Error	Statistic	Std. Error
x1	110	8.50	2.00	10.50	6.1309	2.19389	4.813	.334	.230	-.755	.457
x2	110	8.90	3.10	12.00	6.7100	2.10081	4.413	.289	.230	-.542	.457
x3	110	9.20	2.30	11.50	6.3673	2.32813	5.420	.459	.230	-.763	.457
x4	110	8.40	3.10	11.50	7.9455	1.93099	3.729	-.294	.230	-.481	.457
y1	110	11.80	2.20	14.00	7.0964	2.50317	6.266	.111	.230	-.579	.457
y2	110	11.70	2.30	14.00	7.9082	2.12107	4.499	-.303	.230	.671	.457
y3	110	10.10	2.40	12.50	7.9400	2.02440	4.098	-.340	.230	-.356	.457
y4	110	7.50	2.00	9.50	5.8500	1.61044	2.594	-.047	.230	-.141	.457
y5	110	9.90	2.00	11.90	6.4409	2.25870	5.102	-.123	.230	-.927	.457
y6	110	11.80	2.20	14.00	6.8636	2.48695	6.185	.448	.230	.330	.457
y7	110	10.80	2.20	13.00	8.1664	1.93520	3.745	-.342	.230	.248	.457
y8	110	10.60	2.40	13.00	8.9009	2.82757	7.995	-.484	.230	-.761	.457
y9	110	7.40	2.10	9.50	6.4200	2.12425	4.512	-.240	.230	-1.083	.457
y10	110	9.50	2.50	12.00	7.3545	1.91458	3.666	-.223	.230	-.127	.457
Valid N (listwise)	110										

b. Analisis Statistik Inferensial

1). Uji Asumsi Model

a). Ukuran Sampel

Pada penelitian ini jumlah sampel yang penulis ambil adalah sebesar 110 sampel dari total populasi 1.073 karyawan yang menggunakan Sistem Informasi Karyawan.

b). Uji Normalitas

Hasil pengujian normalitas data yang terdapat pada Tabel 3, dapat dilihat bahwa nilai yang berada pada kolom c.r semuanya berada di dalam *range* yang direkomendasikan, yaitu antara -2,58 sampai 2,58. Oleh karena itu dapat dikatakan bahwa data yang digunakan pada penelitan tersebut setelah diuji normalitas datanya menggunakan *software AMOS for Windows* Versi 16.0.1 ini terdistribusi secara normal secara *univariate*. Sedangkan untuk hasil pengujian normalitas data secara *multivariate* mendapatkan nilai 0,750 (Berada diantara kisaran -2,58 sampai 2,58). Kesimpulannya data yang dipergunakan pada penelitian tersebut terdistribusi secara normal dan dapat dipergunakan serta

memenuhi persyaratan untuk dianalisis lebih lanjut.

c). Uji Outlier

Pengujian *Mahalanobis Distance* dapat dilihat dari hasil keluaran *software AMOS for Windows* versi 16.0.1, pada bagian *Observations farthest from the centroid (Mahalanobis distance)* kemudian *Mahalanobis d-squared*. Pada *Mahalanobis d-squared* terlihat bahwa angka-angka yang tertera pada bagian tersebut berada kisaran < 29,14. Artinya hasil pengujian *Mahalanobis d-squared* yang menyatakan hasil tebaran data yang dihasilkan dari kuesioner masing-masing responden memenuhi persyaratan karena tidak menimbulkan adanya *Multivariate Outlier*. Angka pada *Mahalanobis d-squared* tersebut harus < χ^2 α, df (1%, 14) = 29,14.

Tabel 4. Hasil Pengujian Normalitas Data

Variable	Min	Max	Skew	c.r.	kurtosis	c.r.
y10	2,500	12,000	-,220	-,941	-,175	-,375
Y7	2,200	13,000	-,337	-1,443	,183	,391
X4	3,100	11,500	-,290	-1,242	-,514	-1,100
y8	2,400	13,000	-,477	-2,043	-,781	-1,673
y9	2,100	9,500	-,236	-1,012	-1,089	-2,330
y6	2,200	14,000	,442	1,894	,261	,559
y5	2,000	11,900	-,121	-,520	-,939	-2,011
x1	2,000	10,500	,329	1,410	-,776	-1,660
x2	3,100	12,000	,285	1,222	-,572	-1,224
x3	2,300	11,500	,453	1,939	-,783	-1,676
y4	2,000	9,500	-,046	-,199	-,189	-,405
y3	2,400	12,500	-,335	-1,436	-,394	-,844
y2	2,300	14,000	-,299	-1,281	,586	1,255
y1	2,200	14,000	,109	,468	-,607	-1,300
Multivariate					3,029	,750

d). Multikolinieritas dan Singularitas

Hasil pengolahan data menggunakan *software AMOS for Windows* Versi 16.0.1 pada bagian *Sample Moment*, kemudian *Sample Covariances* menunjukan bahwa nilai *Determinant of sample covariance matrix* = 219656014,343 atau menunjukan angka tidak sama dengan 0. Hasil pengujian tersebut menjelaskan bahwa tidak terdapat masalah multikolinieritas dan singularitas terhadap data yang dianalisis pada pengujian ini.

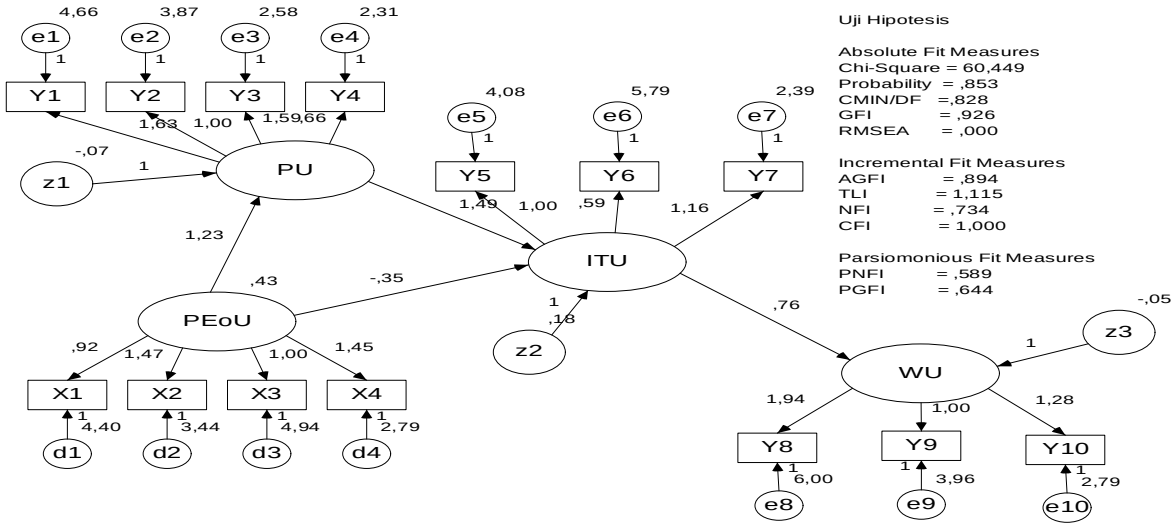
2). Pengolahan dengan Model Persamaan Struktural

a). Objek penelitian

Penelitian ini dilakukan untuk menganalisis variabel Konstruk Eksogen (X) sebagai *source variable* atau *independent variable* yang diprediksi oleh variabel lain. Pada penelitian ini Konstruk Eksogen yang digunakan adalah *Perceive Ease to Use* (PEoU). Sedangkan untuk Konsruk Endogen (Y), adalah faktor yang diprediksi oleh satu atau beberapa konstruk endogen lainnya, tetapi konstruk endogen hanya dapat berhubungan kausal (sebab akibat) dengan konstruk endogen. Pada penelitian ini konstruk endogen yang digunakan adalah *Perceived Usefulness* (PU), *Intention to Use* (ITU) dan *Website Usage* (WU).

b). Pengujian Model Berbasis Teori

Pengujian model berbasis teori penulis lakukan dengan menggunakan *software AMOS for Windows* versi 16.0.1. Hasil dari pengujian model awal yang penulis usulkan dapat dilihat pada gambar berikut :



Gambar 4. Hasil pengujian model awal keseluruhan

Pada Gambar 4 terlihat bahwa model teori atau model awal yang diajukan pada penelitian ini sesuai dengan model populasi yang diobservasi, karena diketahui bahwa nilai probabilitas (P) = 0,853. Hal tersebut sesuai dengan nilai yang direkomendasikan, yaitu probabilitas (P) > 0,05 ((Ghozali 2005), 25). Karena untuk menentukan suatu model dapat dinyatakan sesuai *(fit)* atau tidak hanya dilihat dari nilai probabilitas (P) saja, tetapi ada beberapa persyaratan lain yang harus dipenuhi seperti nilai-nilai *Absolute Fit Measure*, *Incremental Fit Measure* dan *Parsimonius Fit Measures* yang memenuhi batas nilai kritis yang telah ditentukan. Adapun batasan nilai kritis tersebut dapat dilihat pada Tabel 5.

Tabel 5. Batas nilai kritis uji kesesuaian model (Widodo 2006)

Ukuran Kesesuaian	Batas Nilai Kritis	Keterangan
Absolut Fit Measures		
a. Chi-Squre X^2 (CMIN)	Kecil, $\leq \chi 2\ \alpha$; df	(Hulland 1996)
b. Probability	$\geq$ 0,05	(Hulland 1996)
c. Chi-Square X^2 Relatif (CMIN/DF)	$\leq$ 2,0	(Byrne 1988)
d. GFI	$\geq$ 0,90	(Diamondtopaulus 2000)
e. RMSEA	$\leq$ 0,08	(Browne 1993)
Incremental Fit Measures		
a. AGFI	$\geq$ 0,90	(Diamondtopaulus 2000)
b. TLI	$\geq$ 0,95	(Hair 1998)
c. NFI	$\geq$ 0,90	(Bentler 1992)
d. CFI	$\geq$ 0,95	(Arbuckle 1997)
Parsimoniuos Fit Measures		
a. PNFI	$\geq$ 0,60	(James 1992)
b. PGFI	$\geq$ 0,60	(Byrne 1988)

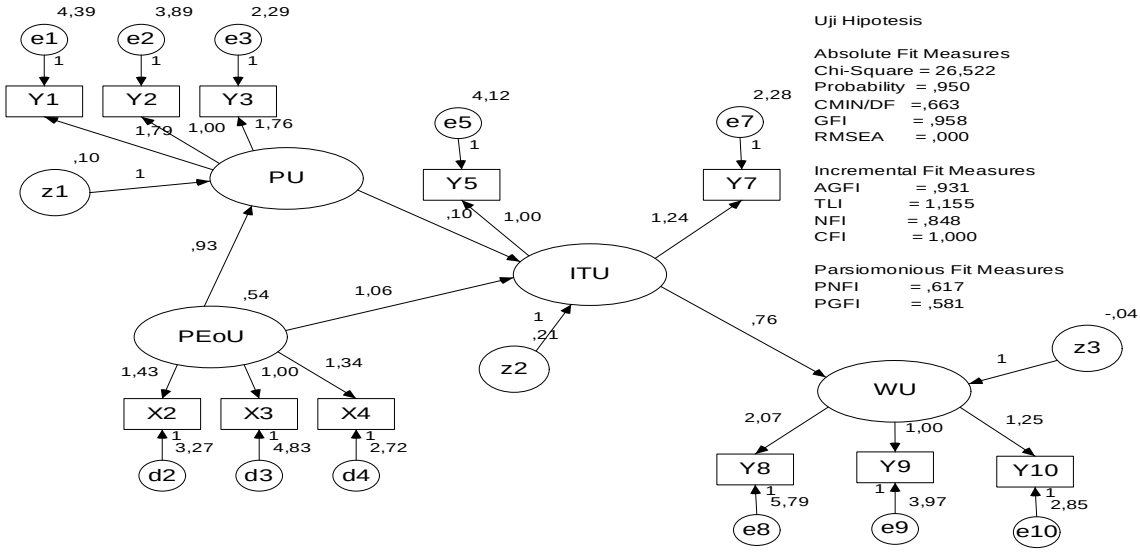
Dengan demikian, kita dapat memodifikasi model yang ada sampai model tersebut dapat dapat dinyatakan sesuai *(fit)*.

Pada penelitian ini penulis menggunakan *Model Developmental Strategy*, artinya strategi ini sangat memungkinkan untuk dilakukannya modifikasi model apabila model yang diajukan belum sesuai *(fit)* sesuai dengan syarat yang direkomendasikan.

Modifikasi dilakukan untuk mendapatkan model yang sesuai *(fit)* dengan persyaratan pengujian (Widodo 2006). Berdasarkan pernyataan teori yang ada, maka dilakukan modifikasi model dengan asumsi perubahan model struktural harus dilandasi

oleh dasar teori yang kuat (Ghozali 2005). Berdasarkan hasil estimasi dan *Regresion Wieght*, maka dilakukan modifikasi dengan menghapus variabel indikator yang bukan merupakan konstruktor yang valid bagi suatu variabel laten pada model struktural yang diajukan. Jika nilai estimasi pada *Loading Factor* (λ) dari suatu variabel indikator < 0,5, maka indikator tersebut hendaknya dihapus (di-*drop*) (Ghozali 2005).

Untuk melihat signifikansi (sig), nilai yang dipersyaratkan adalah < 0,05. Jika nilai signifikansi (sig) > 0,05, maka dapat dikatakan bahwa indikator tersebut bukan merupakan indikator yang *valid* bagi suatu variabel laten dan sebaiknya hal ini dihapus (di-*drop*) (Widodo 2006). Modifikasi dilakukan dengan tujuan untuk mendapatkan nilai probabilitas (P) > 0,05, sehingga model dinyatakan sesuai *(fit)*. Pada penelitian ini, modifikasi dilakukan dalam tiga tahapan. Hasil modifikasi akhir pada pada penelitian ini dapat dilihat pada gambar berikut :



Gambar 5. Hasil modifikasi model tahap ketiga

Berdasarkan hasil pengujian tahap akhir, dapat dilihat nilai probabilitas (P) menunjukan angka = 0,950 dan nilai *Chi Square* (χ^2) = 26,552. Pada tahapan ini, nilai probabilitas (P) dan *Chi Square* (χ^2) sudah dapat dikatakan baik, karena sudah berada pada nilai-nilai yang dipersyaratkan, yaitu probabilitas (P) > 0,05 dan *Chi Square* (χ^2) < 29,14.

c). Uji Kesesuaian Model

Kreteria sesuai *(fit)* atau tidaknya suatu model tidak hanya dilihat dari nilai probalibitas dan, melaikan juga dilihat kretaria lain yang meliputi : ukuran *Absolute Fit Measure*, *Incremental Fit Measures* dan *Parsiomonius Fit Measures*. Untuk membandingkan nilai yang didapat pada model ini dengan batas nilai kritis pada masing-masing kreteria pengukuran tersebut, maka dapat dilihat pada tabel 6.

Tabel 6. Batas nilai kritis uji kesesuaian model

Ukuran Kesesuaian	Batas Nilai Kritis	Hasil Model Ini	Keterangan
Absolut Fit Measures			
a. Chi-Squre X^2 (CMIN)	Kecil, $\leq \chi 2\ \alpha$; df	26,522	Baik
b. Probability	≥ 0.05	0,950	Baik
c. Chi-Square X^2 Relatif (CMIN/DF)	≤ 2.0	0,663	Baik
d. GFI	≥ 0.90	0,958	Baik

e. RMSEA	≤ 0.08	0,000	Baik
Incremental Fit Measures			
a. AGFI	≥ 0.90	0,931	Baik
b. TLI	≥ 0.95	1,155	Baik
c. NFI	≥ 0.90	0,848	Marginal
d. CFI	≥ 0.95	1,000	Baik
Parsimoniuos Fit Measures			
a. PNFI	≥ 0.60	0,617	Baik
b. PGFI	≥ 0.60	0,581	Marginal

Berdasarkan tabel di atas maka dapat dikatakan keseluruhan model dinyatakan sesuai *(fit).* Model yang diajukan pada penelitian ini didukung oleh fakta dilapangan. Hal ini diindikasikan bahwa matriks tersebut varian-kovarians populasi sama dengan matriks varian-kovarians sampel (Data Observasi) atau dapat dinyatakan $\sum_p = \sum_s$.

Pada penelitian ini analisa terhadap model dilakukan dengan dua tahapan, yaitu analisis masing-masing variabel secara sendiri-sendiri *(Confirmatory Factor Analyst)* dan model secara keseluruhan *(Full Model)* yang mengindikasikan bahwa model dinyatakan sesuai *(fit)* secara keseluruhan.

3. Hasil Pengujian

a. Uji Parameter Model Pengukuran Variabel Laten

1). Pengujian Validitas

Pengujian terhadap validitas variabel laten dilakukan dengan melihat nilai signifikansi (Sig) yang diperoleh tiap variabel indikator, kemudian dibandingkan dengan nilai α (0,05). Jika Sig ≤ 0,05 maka tolak H_0, artinya variabel indikator tersebut merupakan konstruktor yang valid bagi variabel laten tertentu.

a). Variabel Laten Eksogen

Pada tabel berikut ini, penulis sajikan hasil pengujian parameter Variabel *Perceived Ease of Use* (PeoU). Hasil pengujian tersebut diambil dari hasil *output software AMOS for Windows* Versi 16.0.1 pada bagian *Estimates* à *Matrices* à *Indirect Effect*.

Tabel 7. Uji Parameter Variabel *Perceived Ease of Use* (PEoU)

***Perceived Ease of Use* (PeoU)**	**Sig (≤ 0.05)**	**Hasil Hipotesis**	**Keterangan**
X2 (Kemudahan untuk dipelajari)	0,000	Tolak H_0	Konstruk yang *valid*
X3 (Kemudahan untuk digunakan)	0,000	Tolak H_0	Konstruk yang *valid*
X4 (Kemudahan untuk berinteraksi)	0,000	Tolak H_0	Konstruk yang *valid*

Masing-masing variabel indikator X2 (Kemudahan untuk dipelajari), X3 (Kemudahan untk digunakan) dan X4 (Kemudahan untuk berinteraksi) secara signifikan merupakan konstruk yang valid bagi variabel laten *Perceived Ease of Use* (PeoU). Dengan demikian respon karyawan terhadap Sistem Informasi Karyawan Berbasis *Web* pada BSI merasa mudah untuk dipelajari, mudah untuk digunakan dan dalam hal berintraksi atau mengaksesnya.

b). Variabel Laten Endogen

- ***Perceived Usefulness* (PU)**

Pada tabel berikut ini, penulis sajikan hasil pengujian parameter Variabel *Perceived Usefulness* (PU). Hasil pengujian tersebut diambil dari hasil *output software AMOS for Windows* Versi 16.0.1 pada bagian *Estimates* à *Matrices* à *Indirect Effect*.

Tabel 8. Uji Parameter Variabel *Perceived Usefulness* (PU)

***Perceived Usefulness* (PU)**	**Sig (≤ 0.05)**	**Hasil Hipotesis**	**Keterangan**
Y1 (Mempertinggi efektivitas)	0,000	Tolak H_0	Konstruk yang *valid*
Y2 (Menjawab kebutuhan informasi)	0,000	Tolak H_0	Konstruk yang *valid*
Y3 (Meningkatkan kinerja)	0,000	Tolak H_0	Konstruk yang *valid*

Masing-masing variabel indikator Y1 (Mempertinggi efektivitas), Y2 (Menjawab kebutuhan informasi) dan Y3 (Meningkatkan kinerja) secara signifikan merupakan konstruk yang *valid* bagi variabel laten *Perceived Usefulness* (PU). Dengan demikian respon karyawan terhadap Sistem Informasi Karyawan Berbasis *Web* pada BSI adalah merasa dapat mempertinggi efektifitas, menjawab kebutuhan informasi dan meningkatkan kinerja.

- ***Intention to Use* (ITU)**

Pada tabel berikut ini, penulis sajikan hasil pengujian parameter Variabel *Intention to Use* (ITU). Hasil pengujian tersebut diambil dari hasil *output software AMOS for Windows* Versi 16.0.1 pada bagian *Estimates* à *Matrices* à *Indirect Effect*.

Tabel 9. Uji Parameter Variabel *Intention to Use* (ITU)

***Intention to Use* (ITU)**	**Sig (≤ 0.05)**	**Hasil Hipotesis**	**Keterangan**
Y5 (Penambahan *software* pendukung)	0,000	Tolak H_0	Konstruk yang *valid*
Y7 (Memotivasi ke pengguna lain)	0,000	Tolak H_0	Konstruk yang *valid*

Masing-masing variabel indikator Y5 (Penambahan *software* pendukung) dan Y7 (Memotivasi ke pengguna lain) secara signifikan merupakan konstruk yang valid bagi variabel laten *Intention to Use* (ITU). Dengan demikian respon karyawan terhadap Sistem Informasi Karyawan Berbasis *Web* pada BSI adalah merasa perlu untuk menambahkan *software* pendukung lainnya dalam menggunakannya, seperti : *software* pen-*download*, anti virus, firewall, dll. Selain itu karyawan perlu memotivasi pengguna lain untuk menggunakan Sistem Informasi Karyawan Berbasis *Web* pada BSI.

- ***Website Usage* (WU)**

Pada tabel berikut ini, penulis sajikan hasil pengujian parameter Variabel *Website Usage* (WU). Hasil pengujian tersebut diambil dari hasil *output software AMOS for Windows* Versi 16.0.1 pada bagian *Estimates* à *Matrices* à *Indirect Effect*.

Tabel 10. Uji Parameter Variabel *Website Usage* (WU)

***Website Usage* (WU)**	**Sig (≤ 0.05)**	**Hasil Hipotesis**	**Keterangan**
Y8 (Memahami cara penggunaan)	0,000	Tolak H_0	Konstruk yang *valid*
Y9 (Menyampaikan kepuasan)	0,000	Tolak H_0	Konstruk yang *valid*
Y10 (Frekuensi Pengguna)	0,000	Tolak H_0	Konstruk yang *valid*

Masing-masing variabel indikator Y8 (Memahami cara penggunaan), Y9 (Menyampaikan kepuasan) dan Y10 (Frekuensi Pengguna) secara signifikan merupakan konstruk yang valid bagi variabel laten *Website Usage* (WU). Dengan demikian responden terhadap Sistem Informasi Karyawan Berbasis *Web* pada BSI adalah merasa sebelum menggunakan menggunakan Sistem Informasi Karyawan Berbasis *Web* pada BSI memahami terlebih dahulu caranya. Responden juga merasa puas dengan kinerja dari Sistem Informasi Karyawan Berbasis *Web* pada BSI. Responden juga menggunakan Sistem Informasi Karyawan Berbasis *Web* pada BSI setiap hari selama bekerja.

2). Pengujian Reliabilitas

a). Pengujian Secara Langsung

Hasil pengujian secara langsung tersebut diambil dari hasil *output software AMOS for Windows* Versi 16.0.1 pada bagian *Estimates* à Scalars à *Squared Multiple Correlations*. Dengan melihat nilai-nilai yang terdapat pada *Square Multiple Correlations* (R^2), reliabilitas dari suatu indikator dapat dilihat dengan mempertahankan nilai R^2 yang menjelaskan seberapa besar proporsi varians

indikator yang dijelaskan oleh variable laten (sedangkan sisanya dijelaskan oleh *Measerment Error*).

Tabel 11. *Square Multiple Correlations* (R^2) untuk variabel X (Eksogen)

Indikator	*Square Multiple Correlations* (R^2)
X2 (Kemudahan untuk dipelajari)	0,752
X3 (Kemudahan untk digunakan)	0,601
X4 (Kemudahan untuk berinteraksi)	0,763

Berdasarkan Tabel 10 dapat dilihat bahwa variabel indikator X4 (Kemudahan untuk berinteraksi) memiliki nilai *Square Multiple Correlations* (R^2) tertinggi, yaitu sebesar 0,763 sehingga dapat disimpulkan bahwa variabel laten *Perceived Ease of Use* (PeoU) berkontribusi terhadap varians X4 sebesar 76,3 %, sedangkan sisanya 23,7 % dijelaskan pada *Measurment Error*.

Tabel 12. *Square Multiple Correlations* (R^2) untuk variabel Y (Endogen)

Indikator	*Square Multiple Correlations* (R^2)
Y1 (Mempertinggi efektivitas)	0,494
Y2 (Meningkatkan efisiensi)	0,328
Y3 (Meningkatkan kinerja)	0,635
Y5 (Penambahan *software* pendukung)	0,385
Y7 (Memotivasi ke pengguna lain)	0,584
Y8 (Memahami cara penggunaan)	0,568
Y9 (Menyampaikan kepuasan)	0,312
Y10 (Frekuensi Pengguna)	0,516

Berdasarkan Tabel 12 dapat dilihat bahwa variabel indikator Y9 (Menyampaikan kepuasan) memiliki nilai *Square Multiple Correlations* (R^2) paling keciul, yaitu sebesar 0,312 %. Artinya dapat disimpulkan bahwa variabel laten *Website Usage* (WU) berkontribusi terhadap varians Y9 sebesar 31,2 %, sedangkan sisanya 68,8 % dijelaskan pada *Measurment Error*.

b). Pengujian Tidak Langsung

Dengan melakukan uji reliabilitas gabungan, pendekatan yang dianjurkan adalah mencari nilai besaran *Composite Reliability* dan *Variance Extraced* dari masing-masing variable laten dengan menggunakan informasi pada *Loading Factor* dan *Measurment Error*. *Composite Reliability* menyatakan ukuran konsistensi internal dari indikator-indikator sebuah konstruk yang menunjukan derajat masing-masing indicator itu mengindikasikan sebuah konstruk/laten yang umum. Sedangkan *Variance Extraced* menunjukan indikator-indikator tersebut telah mewakili secara baik konstruk laten yang dikembangkan (Ghozali 2005). Hasil pengujian secara langsung tersebut diambil dari hasil *output software AMOS for Windows* Versi 16.0.1 pada bagian *Estimates* à Scalars à *Standardized Regression Weights*.

Tabel 13. Hasil Pengujian Reliabilitas

Jenis Pengujian	Valiabel Laten			
	PEoU	PU	ITU	WU
Composite Reliability	0,832	0,719	0,767	0,721
Variance Extraced	0,585	0,502	0,614	0,518

Pada Tabel 13 terlihat bahwa variable *Perceived Ease of Use* (PEoU), *Perceived Usefulness* (PU), *Intention to Use* (ITU) dan *Website Usage* (WU) memiliki nilai *Composite Reliability* ≥ 0,7 dan nilai *Variance Extraced* ≥ 0,5. Hal ini dapat disampaikan seluruh variabel tersebut memenuhi semua batas nilai yang dipersyaratkan agar dapat dikatakan seluruh variabel tersebut realibel dan dapat digunakan untuk penelitian.

b. Uji Parameter Model Struktural

1). Uji Hipotesis

a). Hipotesis Deskriptif

Adapun hipotesis deskriptif dari penelitian ini adalah :

H_1 : Diduga *Perceived Ease of Use* (PeoU) atau persepsi kemudahan karyawan dalam menggunakan Sistem Informasi Karyawan Berbasis *Web* berpengaruh terhadap *Perceived Usefulness* (PU) atau manfaat menggunakan Sistem Informasi Karyawan Berbasis *Web*. Dimana semakin mudah Sistem Informasi Karyawan digunakan, maka akan semakin besar manfaat bagi karyawan yang menggunakannya.

H_2 : Diduga *Perceived Ease of Use* (PeoU) atau persepsi kemudahan karyawan dalam menggunakan Sistem Informasi Karyawan Berbasis *Web* berpengaruh terhadap *Intention to Use (ITU)* atau niat untuk menggunakan Sistem Informasi Karyawan Berbasis *Web*. Dimana semakin mudah Sistem Informasi Karyawan Berbasis *Web* digunakan, maka akan semakin tinggi niat karyawan untuk menggunakannya.

H_3 : Diduga *Perceived Usefulness* (PU) atau manfaat menggunakan Sistem Informasi Karyawan Berbasis *Web* berpengaruh terhadap *Intention to Use* (ITU) atau niat untuk menggunakan Sistem Informasi Karyawan Berbasis *Web*. Dimana semakin besar manfaat penggunaan Sistem Informasi Karyawan Berbasis *Web*, maka akan semakin tinggi niat karyawan untuk menggunakannya.

H_4 : Diduga *Intention to Use* (ITU) atau niat untuk menggunakan Sistem Informasi Karyawan Berbasis *Web* berpengaruh terhadap *Website Usage* (WU) atau perilaku penggunaan Sistem Informasi Karyawan Berbasis *Web* itu sendiri. Dimana semakin besar niat untuk menggunakan Sistem Informasi Karyawan Berbasis *Web*, maka akan semakin tinggi penggunaannya oleh karyawan.

Berdasarkan modifikasi model yang telah dilakukan, terdapat hubungan *Perceived Ease of Use* (PEoU) dengan *Perceived Usefulness* (PU), *Perceived Ease of Use* (PEoU) dengan *Perceived Usefulness* (PU), *Perceived Usefulness* (PU) dengan *Intention to Use* (ITU) dan *Intention to Use* (ITU) dengan *Website Usage* (WU).

b). Hipotesis Statistik

Pengujian hipotesis statistik terdiri dari dua macam pengujian, yaitu :

a). Variabel laten eksogen :

$H_0 : \gamma_n = 0$: Tidak Berpengaruh (Terima H_0)

$H_1 : \gamma_n \neq 0$: Berpengaruh (Tolak H_0)

b). Variabel laten endogen :

$H_0 : \beta_n = 0$: Tidak Berpengaruh (Terima H_0)

$H_1 : \beta_n \neq 0$: Berpengaruh (Tolak H_0)

c). Taraf Nyata

Taraf nyata yang digunakan (α) = 5% atau 0,05

d). Kreteria Pengambilan Keputusan

Adapun kreteri pengambilan keputusan adalah :

1. Jika nilai probabilitas (Sig) > 0,05, maka Terima H_0
2. Jika nilai probabilitas (Sig) < 0,05, maka Tolak H_0

2). Hasil Pengujian Hipotesis

Hasil pengujian hipotesis tersebut diambil dari hasil *output software AMOS for Windows* Versi 16.0.1 pada bagian *Estimates* à Scalars à *Regression Weights*. Adapun hasil pengujian hipotesis tersebut dapat dilihat pada Tabel 13.

Tabel 14. Hasil Pengujian Hipotesis

Hipotesis	Sig (< 0,05)	Hasil Hipotesis
H1 (PeoU à PU)	0,043	Tolak H_0
H2 (PeoU à ITU)	0,038	Tolak H_0
H3 (PU à ITU)	0,041	Tolak H_0
H4 (ITU à WU)	0,013	Tolak H_0

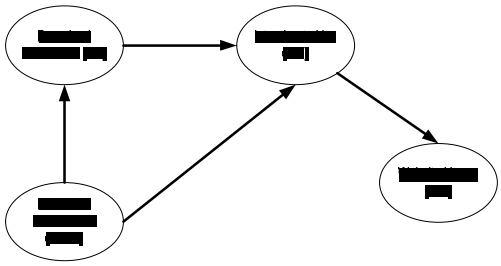
Berdasarkan Tabel 14 di atas, maka dapat dijelaskan sebagai berikut :

1. Variabel *Perceived Ease of Use* (PEoU) memiliki pengaruh terhadap variabel *Perceived Usefulness* (PU).
2. Variabel *Perceived Ease of Use* (PEoU) memiliki pengaruh terhadap variabel *Perceived Usefulness* (PU).
3. Variabel *Perceived Usefulness* (PU) memiliki pengaruh terhadap variabel *Intention to Use* (ITU).
4. Variabel *Intention to Use* (ITU) memiliki pengaruh terhadap variabel *Website Usage* (WU).

4. Interpretasi Model

Berdasarkan hasil modifikasi model dan hasil pengujian hipotes pada penelitian ini,

dapat diketahui bahwa penggunaan Sistem Informasi Karyawan Berbasis *Web* pada BSI dipengaruhi oleh adanya kemudahan atau *Perceived Ease of Use* (PEoU), manfaat menggunakan atau *Perceived Usefulness* (PU), niat untuk menggunakan atau *Intention to Use* (ITU) dan penggunaan *website* atau *Website Usage* (WU). Interpretasi model akhir dari penelitian ini adalah :



Gambar 6. Interpretasi Model Akhir Penelitian

III. PENUTUP

3.1. Kesimpulan

Adapun kesimpulan dari penelitian yang penulis lakukan adalah :

1. Faktor-faktor yang dominan dan saling berhubungan serta berpengaruh terhadap tingkat penerimaan teknologi, khususnya Sistem Informasi Karyawan Berbasis *Web* bagi karyawan kampus BSI adalah :adanya persepsi kemudahan untuk menggunakan *(PEoU)*, *a*danya persepsi manfaat menggunakan (PU), adanya niat untuk menggunakan (ITU) dan penggunaan *website* itu sendiri (WU). Karyawan BSI yang sudah memahami kemudahanmenggunakan Sistem Informasi Karyawan Berbasis *Web* tersebut *(PEoU)* dan manfaat menggunakannya (PU), maka akan mempunyai niat untuk menggunakan Sistem Informasi Karyawan Berbasis *Web* tersebut (ITU).
2. Bentuk model penerimaan sebuah teknologi informasi baru, berupa suatu Sistem Informasi Karyawan Berbasis *Web* yang diterapkan pada BSI adalah PeoU sebagai variabel laten eksogen atau variable independen. PU, ITU dan WU sebagai variabel endogen atau variabel dependen.
3. Model akhir dari penelitian ini sama dengan penelitian dikembangkan dari teori TAM yang diperkenalkan oleh Fred D. Davis pada tahun 1989 yang menguji dua faktor penerimaan teknologi yaitu PU dan PEoU *dan* konsep TAM berdasarkan Money dan Turner pada tahun 2004 (Money 2004). Dimana model tersebut terdiri dari adanya persepsi adanya kemudahan (PEoU), persepsi adanya manfaat menggunakan (PU), niat untuk menggunakan (ITU) dan penggunaan *website* (WU).

3.2. Saran

Adapun saran yang penulis ajukan sesuai dengan hasil penelitian adalah :

1. Mengingat penggunaan Sistem Informasi Karyawan Berbasis *Web* oleh karyawan dipengaruhi oleh faktor-faktor kemudahan menggunakannya, seperti : mudah untuk mengakses, mudah untuk dipelajari atau dipahami, mudah untuk digunakan dan mudah digunakan untuk berinteraksi dengan pengguna lain yang ada di dalam BSI, maka sebaiknya BSI harus lebih mengkonsentrasikan pengembangan Sistem Informasi Karyawan Berbasis *Web* kedepannya lebih mengutamakan hal-hal yang berkaitan dengan kemudahan karyawan untuk menggunakan sistem tersebut.
2. Mengingat penggunaan Sistem Informasi Karyawan Berbasis *Web* oleh karyawan dipengaruhi juga oleh faktor-faktor manfaat menggunakannya, seperti : meningkatkan efektifitas pekerjaan, menjawab kebutuhan akan informasi karyawan, meningkatkan kinerja dan meningkatkan effisiensi, maka sebaiknya BSI juga harus lebih mengkonsentrasikan pengembangan Sistem Informasi Karyawan Berbasis *Web* kedepannya lebih mengutamakan juga hal-hal yang berkaitan dengan manfaat karyawan menggunakan sistem tersebut, seperti : informasi yang ada selalu diperbaharui, menambahkan fasilitas-fasilitas baru yang dapat membantu karyawan dalam menyelesaikan permasalahannya selain fasiitas yang telah tersedia saat ini.
3. BSI diharapkan juga untuk selalu melakukan peninjauan secara berkala, termasuk di dalamnya mengenai umur sistem *(Life Cycle)*, masalah keamanan data, *backup system*, dan hal-hal teknis lainnya yang dapat menggangu keberadaan sistem.
4. BSI diharapkan juga untuk meningkatkan kegunakaan dari Sistem Informasi Karyawan Berbasis *Web* yang ada.
5. Pada penelitian selanjutnya, penulis mengharapkan dapat dikembangkan lagi indikator-indikator atau kisi-kisi pertanyaan yang ada, yang dapat dipergunakan untuk menganalisa lebih dalam dan lebih tepat lagi

mengenai model penerimaan teknologi, khususnya untuk menganalisa model yang tepat untuk penerimaan teknologi berupa Sistem Informasi Karyawan Berbasis *Web*.

DAFTAR PUSTAKA

Davis, Fred D. 1989. *Perceived Usefulness, Perceived Ease of Use and User Accepted of Informations Technology*. Management Information System Quarterly.

Ghozali, Imam. 2003. Model Persamaan Struktural - Konsep dan Aplikasi dengan Program AMOS Ver 5.0. Semarang : BP UNDIP.

Godi, Muhamad Ikbal. 2007. Faktor-faktor yang Mempengaruhi Keberhasilan Penggunaan Knowledge Management System (KMS) : Studi Kasus Penggunaan KAMPIUN Sebagai KMS di PT. Telkom. Tesis. Jakarta : Universitas Budi Luhur.

Hair, J.F. 1998. *Multivariat Data Analyst*, New Jersey : Prentice Hall.

Herlawati. 2007. Faktor-faktor yang Mempengaruhi Perilaku Karyawan dalam Penggunaan Sistem Informasi Perpajakan Berbasis Web Berdasarkan Pendekatan TAM : Studi Kasus di KPP Cikarang Satu – Bekasi. Tesis. Jakarta : Universitas Budi Luhur.

HM, Jogiyanto. 2006. Analisa Desain Sistem Informasi : Pendekatan Terstruktur Teori dan Praktek Aplikasi Bisnis, Edisi Ketiga, Cetakan Kedua, Yogyakarta : Andi.

McLeod, Raymond. 2004. Sistem Informasi Manajemen, Edisi Kedelapan, Jakarta : PT. Indeks.

Money, W., Turner, A. 2004.*Aplications of the Technology Accepted Model to a Knowledge Management System*, In Proceedings of the 37th Hawaii International Conference on System Sciences.

Nasution, Fahmi Natigor. Penggunaan Teknologi Informasi Berdasarkan Aspek Perilaku (Behavioral Aspect), USU Digital Library, http://library.usu.ac.id/download/fe/akuntansi-fahmi2.pdf (Diakses 20 Mei 2008)

Siregar, Syafarudin. 2004. Statistik Terapan untuk Penelitian. Jakarta : Grasindo.

Tanenbaum, Andrew S. 2000. Jaringan Komputer, Edisi Indonesia, Jilid Satu, PT. Prenhallindo dan Pearson Education Asia Pte. Ltd.

Wibowo, Arief. 2006. Kajian Penerapan Sistem Informasi Akademik Berbasis Web Berdasarkan Pendekatan TAM : Studi Kasus di Universitas Budi Luhur, Tesis, Jakarta : Universitas Budi Luhur.

Widodo, Prabowo Pudjo. 2006. *Aplikasi SEM : Management Informatioan System (MIS) & Technology Accepted Model (TAM)*, Jakarta : Universitas Budi Luhur.

_____. 2006. Statistika : Analisis Multivariat, Seri Metode Kuantitatif, Jakarta : Universitas Budi Luhur.

_____. 2006. Petunjuk Pengoperasian AMOS, Jakarta : Universitas Budi Luhur.

Wothke, W. *Nonpositive definite matrices in structural modeling. In Bollen, K.A. & Long, J.S, Testing Structural Equations Models* (pp. 256-293).

MEMPERCEPAT USIA PROYEK MENGGUNAKAN METODE PERT DAN CPM UNTUK PENGEMBANGAN SISTEM APLIKASI KOMPUTER

Rachmat Adi Purnama

AMIK Bina Sarana Informatika
Jl. RS. Fatmawati No. 24 Jakarta Selatan (12450) Indonesia
rachmat_adip@bsi.ac.id

Abstract

At the time a development project application systems are often responsible for the project constraints of time needed to complete the project itself, where the time is set based on the approximate duration of each activity that must be done according to the plan changes from manejement. Based on the problems, then a solution is required to determine the activities which can be complete. The purpose of writing this journal is to provide a solution in overcoming the problems associated with the acceleration time of a project application development system, while the purpose of writing this journal is to apply methods of PERT and CPM in the age of speed or time required in completing a project according to the age plan (UREN). Research methods that do the authors use the method of study literature, and descriptive method by using the table-table operational variables. At the end of the discussion of the results will be described how the age estimates (UPER) can be adjusted to the age of the plan (UREN) long with a new activity, so that the responsible project can implement the project in accordance with the target time is planned by management

Keys : PERT, CPM, UPER, UREN

I. PENDAHULUAN

Pembangunan atau pengembangan sebuah sistem informasi dalam sebuah perusahaan membutuhkan waktu penyelesaian yang harus sudah ditentukan. Hal ini tentunya menjadi tanggung jawab seorang pimpinan proyek untuk dapat menyelesaikan proyek pembangunan atau pengembangan sistem sesuai dengan waktu yang telah ditentukan tersebut.

Pada pelaksanaanya kadang pimpinan proyek mengalami kendala waktu penyelesaian dikarenakan adanya perubahan waktu yang telah ditentukan dengan kebijakan manajemen yang menginginkan agar proyek tersebut dipercepat penyelesainya. Oleh karena itu, maka langkah yang harus dilakukan oleh penanggung jawab proyek adalah menentukan kegiatan apa saja yang dapat dipercepat kegiatannya tanpa harus mempengaruhi kegiatan-kegiatan lainnya.

Beberapa metode mempercepat umur proyek dapat digunakan untuk mengatasi permasalahan di atas. Pada penulisan jurnal ini penulis mencoba menggunakan metode PERT dan CPM yang merupakan konsep analisa jaringan. Metode ini dapat memperlihatkan kegiatan-kegiatan mana saja yang nantinya dapat dipercepat lama kegiatannnya.

Tujuan Penelitian

1. Mengetahui berapa toleransi waktu kegiatan yang dibutuhkan untuk masing-masing kegiatan
2. Menentukan kegiatan mana saja yang dapat dipercepat waktu penyelesaiannya sesuai dengan umur rencana (UREN)
3. Menentukan jalur kritis pada pengerjaan proyek dengan menggunakan metode PERT dan CPM, sehingga diketahui berapa umur perkiraan (UPER) penyelesaian pengembangan proyek sistem aplikasi itu sendiri.
4. Menentukan lama kegiatan baru masing-masing kegiatan sesuai dengan umur rencana (UREN)

Pembahasan penulisan jurnal ini penulis batasi berdasarkan waktu kegiatan yang telah ditentukan tanpa biaya dan sumberdaya yang dibutuhkan, dimulai dari proses pembuatan tabel kegiatan yang telah diketahui lama perkiraan masing-masing kegiatan, proses pembuatan *network* diagram PERT dan CPM berdasarkan umur perkiraan (UPER) sampai dengan pembuatan tabel kegiatan dan network diagram PERT dan CPM dengan lama kegiatan baru sesuai dengan umur rencana (UREN) yang telah ditentukan.

II. PEMBAHASAN

TINJAUAN PUSTAKA

Menurut Subagyo dan Pangestu (2000), analisa *network* biasa dikenal dengan nama teknik manajemen proyek. Kebutuhan penyusunan *network* ini dirasakan perlu karena adanya koordinasi dan pengurutan kegiatan–kegiatan yang kompleks, yang saling berhubungan dan saling tergantung satu sama lain. Menurut Soepranto (2001), CPM mulai dikembangkan tahun 1957 oleh J.E.Kelly dari Remington Rand dan M.R.Walker dari DuPont, dan PERT mulai dikembangkan tahun 1958 oleh Booz, Allen, dan Hamilton. Kedua teknik ini dikembangkan untuk membantu para manajer membuat penjadwalan, memonitor, dan mengendalikan proyek besar dan kompleks.

Pengertian menurut Heizer dan Barry Render (2005) PERT adalah Untuk membagi keseluruhan proyek ke dalam kejadian dan aktivitas. Suatu kejadian menandai mulainya atau selesainya tugas atau aktivitas tertentu. Suatu aktivitas di sisi lain adalah suatu tugas atau subproyek yang terjadi antara dua kejadian. Menurut Heizer dan Render (2005), dalam jaringan PERT kita menetapkan tiga perkiraan waktu (*three times estimates*) untuk masing-masing jaringan aktivitas.

Menurut Heizer dan Render (2005), kelebihan PERT :

1. Sangat berguna terutama saat menjadwalkan dan mengendalikan proyek besar.
2. Konsep yang lugas atau secara langsung (*straightforward*) dan tidak memerlukan perhitungan matematis yang rumit.
3. Jaringan grafis membantu melihat hubungan antar kegiatan secara cepat.
4. Analisis jalur kritis dan waktu *slack* membantu menunjukkan kegiatan yang perlu diperhatikan lebih dekat.
5. Dokumentasi proyek dan gambar menunjukkan siapa yang bertanggung jawab untuk kegiatan yang beragam.
6. Dapat diterapkan untuk proyek yang bervariasi.
7. Berguna dalam mengawasi jadwal dan biaya.

Menurut Heizer dan Render (2005), keterbatasan dalam PERT :

1. Kegiatan proyek harus ditentukan secara jelas, dan hubungannya harus bebas dan stabil.
2. Hubungan pendahulu harus dijelaskan dan dijaringkan bersama-sama.
3. Perkiraan waktu cenderung *subjektif* dan bergantung pada kejujuran para manajer yang takut akan bahaya terlalu optimistis atau tidak cukup pesimistis.
4. Ada bahaya terselubung dengan terlalu banyaknya penekanan pada jalur terpanjang atau kritis. Jalur yang nyaris kritis perlu diawasi dengan baik.

Menurut Retno (2005) penggabungan metode PERT dan CPM digunakan untuk menyusun jadwal yang dihasilkan dari output metode PERT dengan menggunakan CPM. Begitu solusi dengan metode PERT telah diperoleh, langkah selanjutnya adalah menentukan bagaimana karakteristik jadwal yang diinginkan. Jadwal yang disusun dapat dioptimumkan menurut kebutuhan dan kendala sumber daya yang ada.

A. Metode CPM (*Critical Path Method)*

Disamping PERT, Metode Jalur Kritis (CPM) menurut Tubagus (1997) merupakan metode untuk merencanakan dan mengendalikan proyek-proyek, adalah sistem yang paling banyak digunakan diantara semua sistem lain yang memakai prinsip pembentukan jaringan.

Prinsip-prinsip pembentukan jaringan dalam CPM mirip sekali dengan prinsip-prinsip dalam sistem PERT. Jadi mereka yang sudah mengenal PERT dengan baik, tidak menemui kesulitan lagi dalam mempergunakan CPM, sejauh hal ini menyangkut pembentukan jaringannya. Perbedaan utama terletak dalam penentuan perkiraan waktunya.

Menurut Tubagus (1997) dalam sistem CPM ditentukan dua buah perkiraan waktu dan biaya untuk setiap aktifitas yang terdapat dalam jaringan. Kedua perkiraan ini adalah perkiraan normal (*Normal Estimate*) dan perkiraan cepat (*Crash Estimate*). Perkiraan waktu yang normal kira-kira sama dengan perkiraan waktu yang paling mungkin dalam PERT. Biaya normal tentu saja adalah biaya yang diperlukan untuk menyelesaikan suatu proyek dalam waktu normal. Perkiraan waktu cepat adalah waktu yang akan dibutuhkan oleh suatu proyek jika biaya yang dikeluarkan tidak jadi soal dalam

usaha mempersingkat waktu bagi proyek tersebut. Dalam program semacam ini, manajer akan melakukan segala apa saja yang dirasakan perlu untuk mempercepat selesainya pekerjaan. Jadi biaya mempercepat adalah biaya yang dibutuhkan untuk melaksanakan sesuatu pekerjaan yang dipercepat selesainya, denga tujuan untuk mempercepat waktu selesainya secepat mungkin.

Perbandingan PERT versus CPM dapat dilihat pada tabel 1.

Tabel 1. Perbandingan PERT versus CPM

No	Fenomena	CPM	PERT
1	Estimasi kurun waktu kegiatan	Deterministik, satu angka	Probabilistik, tiga angka
2	Arah orientasi	Ke kegiatan	Ke Peristiwa/kejadian
3	Indentifikasi jalur kritis	Dengan hitungan maju dan mundur	Sama dengan CPM
4	Kurun waktu penyelesaian milestone/proyek	Ditandai dengan suatu angka tertentu	Angka tertentu ditambah varians
5	Kumungkinan (probability) mencapai target jadwal	Hitungan/analisis untuk maksud tersebut tidak ada	Angka tertentu ditambah varians
6	Menganalisa jadwal yang ekonomis	Prosedurnya jelas	Mungkin perlu dikonversikan ke CPM terlebih dahulu

Sumber :
Studi kelayakan Proyek Industri Soeharto (2002)

B. Penentuan lama kegiatan

Untuk pekerjaan-pekerjaan standar, biasanya telah tersedia suatu standar yang menentukan hubungan antara volume pekerjaan, sumberdaya yang tersedia, dan waktu, sehingga menentukan hari kerja untuk pekerjaan yang bersangkutan bukan merupakan persoalan lagi.

Menurut Tubagus (1997), jika belum tersedia standar yang telah ditetapkan, maka terdapat tiga waktu perkiraan yang dapat digunakan untuk menentukan lama kegiatan antara lain :

1. Waktu optimis/*optimistic time* (LO) : Waktu terpendek kejadian yang mungkin terjadi. Waktu yang dibutuhkan oleh sebuah kegiatan jika semua hal berlangsung sesuai rencana. Dalam memperkirakan nilai ii, biasanya terdapat peluang kecil (katakanlah, 1/100) bahwa waktu kegiatan akan < ***a*** .
2. Waktu pesimis/*pessimistic time* (LP) : Waktu terpanjang kejadian yang dibutuhkan. Waktu yang dibutuhkan sebuah kegiatan dengan asumsi kondisi yang ada sangat tidak diharapkan. Dalam memperkirakan nilai ini, biasanya terdapat peluang yang juga kecil (juga, 1/100) bahwa waktu kegiatan akan > ***b*** .
3. Waktu realistis/*most likely time* (LM) : Waktu yang paling tepat untuk penyelesaian aktivitas dalam jaringan PERT, merupakan waktu yang paling sering terjadi jika suatu aktivitas diulang beberapa kali.

Penentukan lama waktu perkiraan (LPER) dapat menggunakan rumus Diminta :

(1)

$$LPER = \frac{1 * LO + 4 * LM + 1 * LP}{6}$$

C. Saat Paling Awal (SPA)

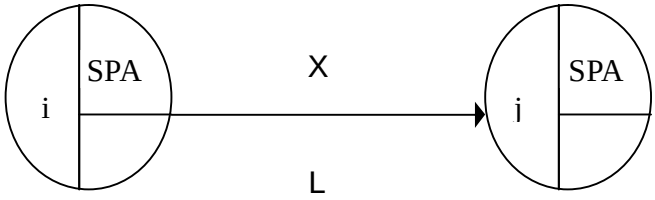
Menurut Tubagus (1997) Saat paling awal (SPA) adalah saat paling awal suatu peristiwa mungkin terjadi, dan tidak mungkin terjadi sebelumnya. Penentuan SPA ini bertujuan untuk mengetahui saat paling awal mulai melaksanakan kegiatan-kegiatan yang keluar dari peristiwa yang bersangkutan.

Menurut Tubagus (1997) terdapat beberapa syarat yang harus dipenuhi untuk menentukan atau menghitung saat paling awal semua peristiwa-peristiwa pada sebuah network diagram antara lain :

1. Network diagram yang tepat telah tersedia.
2. Nomor-nomor peristiwa ditetapkan menurut atau memenuhi persyaratan yaitu peristiwa awal network diagram diberi nomor 1, peristiwa akhir diberi nomor maksimum yang sama dengan banyaknya peristiwa yang ada di network yang bersangkutan.
3. Semua kegiatan yang ada dalam network diagram telah ditetapkan lama kegiatan perkiraan.

Secara formatif untuk menentukan saat paling awal suatu peristiwa adalah sebagai berikut :

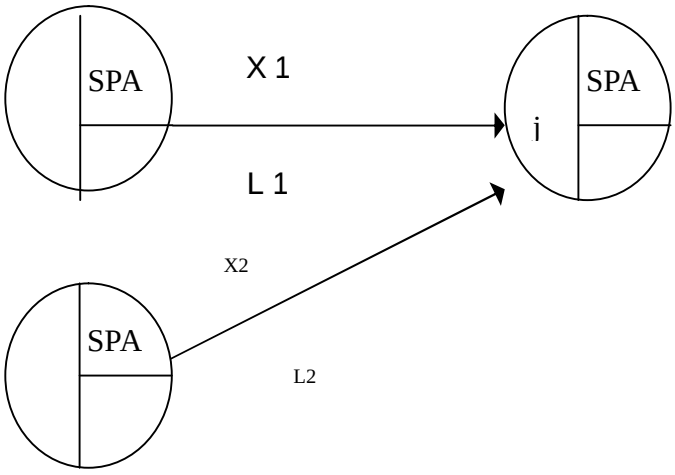
1. Sebuah kegiatan menuju ke sebuah peristiwa



Keterangan :
SPA j = SPA I + L
X = Kegiatan
j = Peristiwa akhir kegiatan X
i = Peristiwa awal kegiatan X
L = Lama kegiatan X yang diperkirakan.
SPA i = Saat paling awal peristiwa awal
SPA j = Saat paling awal peristiwa akhir

Gambar 1. Menentukan saat paling awal satu peristiwa

2. Untuk beberapa kegiatan menuju ke sebuah peristiwa ditentukan waktu paling awal yang maksimum



Keterangan :
SPA j = maksimum (SPA I + L)

Gambar 2. Menentukan saat paling awal beberapa kegiatan

Prosedur untuk menghitung saat paling awal dalam sebuah network diagram adalah sebagai berikut :

1. Hitung atau tentukan saat paling awal dari peristiwa-peristiwa mulai dari nomor 1 berturut-turut sampai dengan nomor maksimum.
2. Pada posisi awal SPA (1) = 0
3. Pada setiap posisi j , SPA j = max (SPAi + L), untuk setiap posisi atau peristiwa i yang dihubungkan langsung oleh satu kegiatan dengan posisi j

D. Saat paling lambat (SPL)

Menurut Tubagus (1997:64) saat paling lambat (SPL) adalah saat paling lambat suatu peristiwa boleh terjadi, dan tidak boleh sesudahnya (meskipun itu mungkin) sehingga proyek mungkin selesai oada waktu yang telah direncanakan.

Menurut Tubagus (1997) terdapat beberapa syarat harus yang harus dipenuhi agar bisa menentukan atau menghitung saat paling lambat (SPL) semua peristiwa-peristiwa pada sebuah network diagram adalah :

1. Telah tersedianya network diagram yang tepat
2. Telah dihitung saat paling awal (SPA) dari posisi atau peristiwa awal (nomor 1) sampai dengan posisi akhir pada network diagram tersebut.
3. Jika hanya ada sebuah kegiatan keluar dari sebuah peristiwa, maka SPL (saat paling Lambat) peristiwa tersebut dalah saat paling lambat mulainya kegiatan tersebut.

Secara formulatif untuk menentukan saat paling lambat suatu peristiwa adalah :

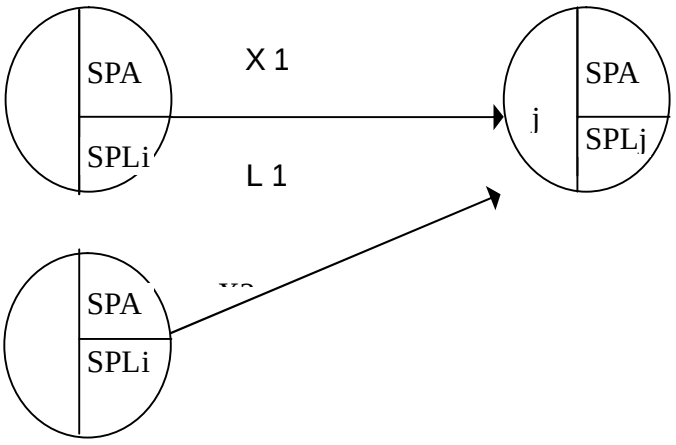
1. Untuk sebuah kegiatan keluar dari sebuah peristiwa

Keterangan :

SPL i = SPL j - L
X = Kegiatan
j = Peristiwa akhir kegiatan X
i = Peristiwa awal kegiatan X
L = Lama kegiatan X yang diperkirakan
SPA i = Saat paling awal peristiwa awal
SPA j = Saat paling awal peristiwa akhir
SPL i = Saat paling lambat peristiwa awal
SPL j =Saat paling lambat peristiwa akhir

Gambar 3. Menentukan saat paling lambat satu peristiwa

2. Untuk beberapa kegiatan keluar dari sebuah peristiwa ditentukan waktu paling lambat yang minimum.



L2

Keterangan

SPL i = minimum (SPL j- L)

Gambar 4. Menentukan saat paling lambat beberapa kegiatan

Prosedur untuk menghitung saat paling awal dalam sebuah network diagram adalah sebagai berikut :

1. Hitung atau tentukan saat paling lambat (SPL) dari peristiwa-peristiwa mulai dari nomor maksimal kemudian mundur berturut-turut sampai dengan posisi awal.
2. Saat paling lambat (SPL) sama dengan Saat paling awal peristiwa akhir (maksimal), SP L j = SPA j
3. Pada setiap posisi i , SPL i = min (SPL j - L), untuk setiap posisi atau peristiwa j yang dihubungkan langsung oleh satu kegiatan dengan posisi i
4. Pada posisi awal SPL 1 = 0

Sedangkan Umur Perkiraan proyek (UPER) ditentukan oleh saat paling awal kegiatan yang paling awal mulai dikerjakan, yaitu SPA peristiwa awal network diagram, dan ditentukan oleh saat paling awal kegiatan akhir yang paling akhir selesai, yaitu SPA peristiwa akhir network diagram dengan syarat SPA awal network diagram sama dengan nol.

METODE PENELITIAN

Pada dasarnya metode penelitian merupakan cara ilmiah untuk mendapatkan data dengan tujuan dan kegunaan tertentu. Penulis menggunakan dua buah metode yaitu metode studi pustaka dengan membaca beberapa sumber literatur baik dari buku-buku penunjang maupun dari beberapa jurnal ilmiah yang terdapat di internet. Sedangkan metode yang kedua yaitu metode penelitian deskriptif, yaitu dimana penulis melakukan survei terhadap dokumentasi maupun data-data yang dibutuhkan dari beberapa kasus pembangunan maupun pengembangan sistem aplikasi komputer dengan tujuan untuk memperoleh gambaran tertentu mengenai suatu masalah yang aktual berdasarkan informasi dan data yang telah ada dan ditarik kesimpulan.

Berdasarkan permasalahan yang ada, maka penulis menggunakan metode deskiriptif terhadap tabel-tabel yang dihasilkan dari hasil penelitian. Adapun operationalisasi variabel sebagai berikut :

Tabel 2. operasional variabel

No	Variabel	Indikator	Definisi Indikator
1	waktu	Saat Paling Awal (SPA)	Yaitu saat paling awal suatu peristiwa mungkin terjadi, dan tidak mungkin terjadi sebelumnya (Rachmat 1997)
2	Waktu	Saat Paling Lambat (SPL)	Yaitu saat paling lambat suatu peristiwa boleh terjadi, dan tidak boleh sesudahnya (meskipun itu mungkin), sehingga proyek mungkin selesai pada waktu yang telah direncanakan (Rachmat 1997)
3	waktu	Lama perkiraan (LPER)	Yaitu waktu perkiraan masing-masing kegiatan berdasarkan perhitungan

			waktu pesimis, waktu optimis dan waktu yang sering terjadi (Rachmat 1997)
4	Waktu	Umur Perkiraan(UPER)	Yaitu umur perkiraan proyek berdasarkan lintasan yang terbentuk mulai dari awal peristiwa sampai akhir peristiwa (Rachmat 1997)
5	Waktu	Umur rencana (UREN)	Yaitu umur rencana proyek berdasarkan kesepakatan antara penanggung jawab proyek dan fihak managemen perusahaan dengan tujuan untuk mempercepat selesainya proyek (Rachmat 1997)
6	Waktu	Lintasan kritis	Yaitu lintasan yang terdiri dari kegiatan-kegiatan kritis mulai dari awal peristiwa sampai dengan akhir peristiwa dan dapat digunakan untuk menentukan umur proyek (Rachmat 1997)

HASIL DAN PEMBAHASAN

Pada penyelenggaran suatu proyek sering kali dihadapkan adanya perbedaan antara umur perkiraan (UPER) berdasarkan *network* diagram yang dibuat dengan umur rencana (UREN) proyek yang ditentukan berdasarkan kebutuhan manajemen dan atau sebab lainnya. Oleh karena itu perlu kiranya umur perkiraan (UPER) dan umur rencana (UREN) harus disamakan. Umur rencana (UREN) biasanya selalu lebih kecil dari umur perkiraan (UPER).

Dengan menggunakan metode PERT dan CPM hal tersebut dapat diatasi, namun harus mengikuti beberapa syarat yang terlebih dahalu antara lain :

1. Telah ada *network* diagram yang tepat.
2. Lama kegiatan perkiraan masing-masing kegiatan telah ditentukan.
3. Telah dihitung saat paling awal (SPA) dan saat paling lambat (SPL) semua peristiwa.
4. Ditentukan umur rencana (UREN).

Setelah memenuhi syarat yang telah ditentukan, maka prosedur berikutnya untuk mempercepat usia proyek adalah :

1. Buat network diagram dengan nomor-nomor peristiwa sama seperti semula dengan lama kegiatan perkiraan baru untuk langkah ulangan, dan sama dengan semula untuk langkah siklus pertama.
2. Dengan dasar saat paling awal peristiwa awal, SPA 1 = 0, dihitung saat peristiwa lainnya. Umur perkiraan proyek (UPER) = saat paling awal peristiwa akhir (SPA m , dimana m adalah nomor peristiwa akhir network diagram atau nomor maksimal peristiwa).
3. Dengan dasar saat paling lambat peristiwa akhir network diagram (SPL m) = umur proyek yang direncanakan (UREN), dihitung saat paling lambat semua peristiwa.
4. Hitung Total Float (TF) semua kegiatan yang ada. Bila tidak ada total float (TF) yang berharga negatif, proses perhitungan selesai. Bila masih ada total float (TF) berharga negatif, lanjutkan ke langkah ke lima
5. Cari lintasan atau lintasan-lintasan yang terdiri dari kegiatan-kegiatan yang Total Floatnya (TF) masing-masing sebesar : Total float (TF)

(2)

= UREN -UPER
= SPL m - SPA m } berharga negatif
= SPL 1 - SPA 1

6. Lama kegiatan dari kegiatan tersebut di atas adalah L n, n adalah nomor urut kegiatan tersebut dalam satu lintasan, n = 1, 2, 3 , ...z.
7. Hitung lama kegiatan baru dari kegiatan tersebut di atas (langkah ke 5 dan ke 6) dengan menggunakan rumus :

(3)

$$L\,n\,(baru) = Ln\,(lama) + \frac{L\,n\,(lama)}{Li} \times (UREN - UPER)$$

Keterangan :
L n (baru) = Lama kegiatan baru
L n (lama)= Lama kegiatan lama
L i = Jumlah lama kegiatan-kegiatan pada satu lintasan yang harus dipercepat.
UREN = Umur rencana proyek
UPER = Umur perkiraan proyek
8. Kembali ke langkah satu

Berdasarkan kegiatan-kegitan yang telah ditentukan dalam pengembangan sistem aplikasi komputer, maka berikut ini adalah tabel aktifitas kegiatan dengan waktu perkiraan yang telah ditentukan beserta kegiatan pendahulunya. Pada proyek pengembangan sistem aplikasi komputer ini, telah ditentukan kegiatan-kegiatan standar dengan lama perkiraan (LPER) yang telah ditentukan terlebih dahulu. Masing-masing kegiatan diberi kode kegiatan, namun tidak secara urut untuk pengerjaannya. Kegiatan pendahulu (*Predesessor*) merupakan kegiatan yang harus diselesaikan terlebih dahulu sebelum kegiatan tersebut dilaksanakan.

Tabel 3. Aktifitas kegiatan beserta waktu perkiraan

NO	Nama Kegiatan	Kode Kegiatan	Waktu perkiraan (LPER)	*PREDESESSOR*
1	Analisa dan disain kebutuhan	A	5	-
2	Pembelian perangkat keras	B	5	-
3	Pembuatan program	C	20	A
4	Instalasi Sistem	D	5	B
5	Test Kode	E	15	C,D
6	Pembuatan buku panduan	F	15	D
7	Konversi	G	5	D
8	Test Sistem	H	5	E
9	Pelatihan	I	5	F
10	User test	J	5	E,I,G

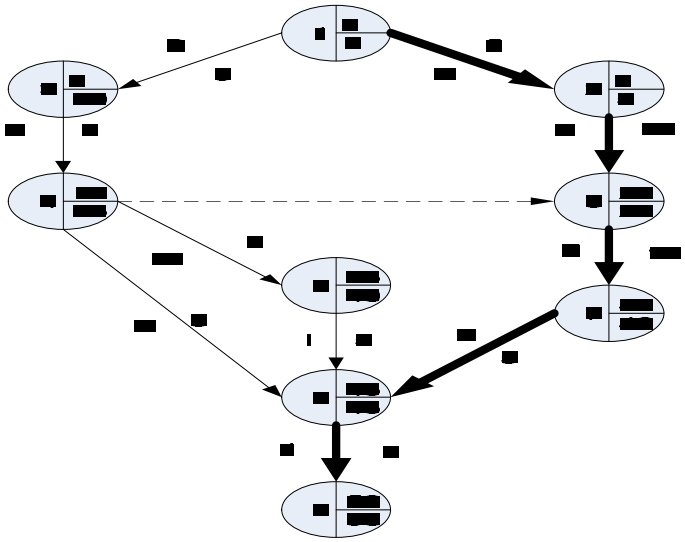
Penjelasan kegiatan :
1. Analisa dan disain kebutuhan
   Pada kegiatan ini dilakukan beberapa aktifitas kegiatan analisa kebutuhan sistem beserta permasalahan dan pemilihan alternatif pemecahannya. Setelah proses analisa, maka aktifitas selanjutnya adalah melakukan disain kebutuhan yang bertujuan untuk mempersiapkan proses implementasi pembuatan program oleh programmer dan pembuatan spesifikasi dan design software. Aktifitas kegiatan analisa dan disain kebutuhan ini diperkirakan membutuhkan waktu lima hari.
2. Pembelian perangkat keras
   Aktifitas kegiatan pembelian perangkat keras sebagai penunjang sistem dapat dilakukan bersama-sama dengan kegiatan analisa dan disain kebutuhan. Pada kegiatan ini dilakukan survai maupun observasi mengenai supplier maupun harga yang berlaku saat ini. Waktu yang dibutuhkan pada kegiatan pembelian perangkat keras ini diperkirakan membutuhkan waktu lima

hari.

3. Pembuatan program
   Proses pembuatan program dapat dilakukan setelah kegiatan analisa dan disain kebutuhan sudah selesai dilaksanakan. Kegiatan ini nantinya akan dilakukan oleh program dengan mengacu kepada disain sistem yang telah dibuat oleh analis sistem. Waktu yang dibutuhkan pada kegiatan ini diperkirakan membutuhkan waktu 20 hari.
4. Instalasi sistem
   Kegiatan instalasi sistem dilakukan setelah seluruh perangkat keras yang dibutuhkan telah tersedia. Adapun kegiatan yang dilakukan pada proses instalasi sistem ini diantaranya adalah melakukan instalasi sistem operasi yang digunakan beserta instalasi jaringan komunikasi data. Waktu yang dibutuhkan pada kegiatan ini diperkirakan membutuhkan waktu lima hari.
5. Test kode
   Kegiatan test kode ini dapat dilakukan setelah kegiatan pembuatan program dan instalasi sistem telah selesai dilaksanakan. Pada kegiatan ini kode-kode yang nantinya akan digunakan dilakukan proses uji coba, sehingga akan diketahui permasalahan maupun kendala yang terjadi sebelum dilakukan proses konversi. Waktu yang dibutuhkan pada kegiatan ini diperkirakan membutuhkan waktu 15 hari.
6. Pembuatan buku panduan
   Pembuatan buku panduan penggunaan sistem sangat diperlukan untuk memudahkan user dalam implemetasi sistem nantinya. Kegiatan ini dilakukan setelah proses instalasi selesai dilakukan. Adapun waktu yang dibutuhkan dalam kegiatan ini diperkirakan membutuhkan waktu l5 hari.
7. Konversi
   Aktifitas konversi ini dilakukan untuk memindahkan data-data yang dibutuhkan oleh sistem ke dalam program yang telah selesai dibuat. Berdasarkan hasil konversi ini akan diketahui bagaimana proses *input* data, proses pengolahan data maupun *output* yang akan dihasilkan. Waktu yang dibutuhkan pada kegiatan ini diperkirakan lima hari.
8. Test Sistem
   Test sistem digunakan untuk mengetahui jalannya sistem maupun kendala-kendala yang dihadapi. Hasil test sistem ini nantinya akan menjadi tolak ukur kelayakan penggunaan sistem yang sedang dibangun atau dikembangkan. Waktu yang dibutuhkan pada kegiatan ini diperkirakan membutuhkan waktu lima hari, dan dilaksanakan setelah kegiatan test kode selesai.
9. Pelatihan
   Kegiatan pelatihan sangat dibutuhkan bagi pengguna sistem agar nantinya tidak menemui kendala teknis, dan mengetahui cara penggunaan sistem baru tersebut. Kegiatan ini dilakukan setelah kegiatan test sistem selesai dilaksanakan. Waktu yang dibutuhkan untuk kegiatan pelatihan ini diperkirakan membutuhkan waktu selama lima hari
10. *User test*
    Untuk mengetahui keberhasilan sistem yang baru, maka kegiatan berikutnya adalah dilakukan ujicoba oleh pengguna. Kegiatan ini dilakukan setelah kegiatan *test kode*, pelatihan dan konversi selesai dilaksanakan. Waktu yang dibutuhkan untuk kegiatan *User test* ini diperkirakan membutuhkan waktu lima hari.

Berdasarkan tabel kegiatan yang telah dibuat langkah berikutnya adalah penulis akan memberikan gambaran bagaimana teknik mempercepat usia proyek dengan menggunakan teknik PERT atau CPM. Adapun waktu perkiraan yang digunakan nantinya akan disesuaikan dengan waktu atau usia rencana (UREN) yang diinginkan oleh fihak penanggung jawab proyek.

Berikut ini adalah gambar PERT dan CPM dengan usia proyek yang dibutuhkan selama 50 hari :



Keterangan :
SPA 1 = Saat paling awal peristiwa awal proyek = 0
SPL 1 = Saat paling lambat peristiwa awal proyek = 0
SPA m = Saat paling awal peristiwa akhir proyek = 50
SPL n = Saat paling lambat peristiwa akhir proyek = 50
UPER = Umur perkiraan proyek yang dibutuhkan = 50
Lintasan kritis = A-C-E-H-J

Gambar 5. PERT/CPM dengan waktu kegiatan lama

Berdasarkan gambar di atas maka dapat diketahui usia atau umur perkiraan proyek yaitu 50 hari. Usia proyek tersebut dapat dilihat berdasarkan kepada kumpulan kegiatan-kegiatan kritis yaitu kegiatan A-C-E-H-J. Kegiatan kristis tersebut nantinya akan membentuk sebuah lintasan kritis, dimana lintasan tersebut berisi kegiatan-kegiatan kritis yang tidak mempunyai toleransi keterlambatan. Untuk mengetahui kegiatan-kegiatan yang mempunyai toleransi keterlambatan dan memiliki Total Float dapat di lihat pada tabel berikut :

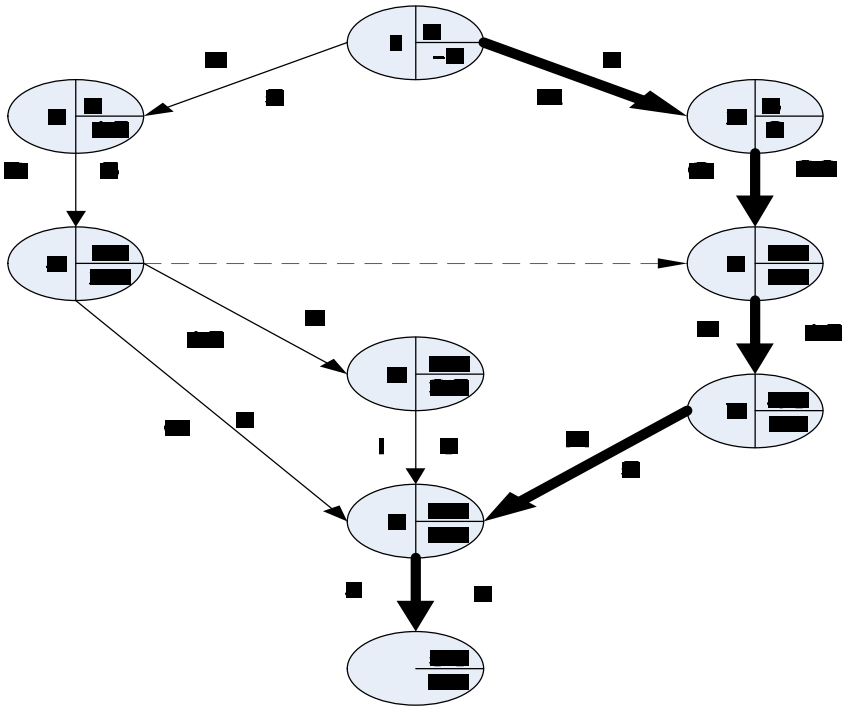
Tabel 4. Jadwal kegiatan dengan Total Float (TF)

Kode Kegiatan	SPL j	Ln	SPA I	TF
A	5	5	0	0
B	20	5	0	15
C	25	20	5	0
D	25	5	5	15
E	40	15	25	0
F	40	15	10	15
G	45	5	10	30
H	45	5	40	0
I	45	5	25	15
J	50	5	45	0

Pada tabel 3. dapat terlihat saat paling lambat (SPLj) dan saat paling awal (SPAi) sebuah kegiatan, beserta lama waktu yang dibutuhkan oleh masing-masing kegiatan. Dengan menggunakan rumus :
TF = SPLj – L – SPA i , maka dapat diketahui berapa toleransi masing-masing kegiatan.

Kadang-kadang sering terjadi sebuah proyek mengalami perubahan umur atau usia proyek yang dibutuhkan, dikarena suatu faktor adanya kebijakan manajemen atau faktor lain yang menyebabkan terjadinya perubahan tersebut. Untuk mengatasi masalah tersebut, maka perlu ditentukan umur rencana (UREN) yang nantinya dapat digunakan sebagai dasar mengurangi lama kegiatan yang dapat memenuhi UREN tersebut.

Pada metode PERT dan CPM langkah yang dapat dilakukan adalah dengan membuat network diagram, dimana usia proyek menjadi UREN pada SPLj nya dan SPL masing-masing kegiatan akan dihitung berdasarkan SPLj akhir sampai dengan SPLi awal peristiwa. Penggambaran network diagram dengan UREN yang telah ditentukan misalkan sebesar 45 hari. Adapun penentuan UREN tersebut dianggap sebagai kebijakan dari fihak manajemen yang menginginkan proyek tersebut dipercepat, maka akan didapat gambar PERT dan CPM sebagai berikut :



Keterangan
UREN = Umur rencana proyek =45
UPER = Umur perkiraan proyek =50
SPA 1 = Saat paling awal peristwa awal proyek = 0
SPA m = Saat paling awal peristiwa akhir proyek = 50
SPL m = Saat paling lambat peristiwa akhir proyek =45

Gambar 6. PERT dan CPM dengan UREN =45

Berdasarkan gambar PERT dan CPM di atas, dapat dilihat bahwa terjadi perubahan usia proyek sebesar -5 dengan mengurangi antara umur perkiraan (UPER) – Umur Rencana (UREN). Kemudian langkah berikutnya yaitu dengan melakukan perhitungan kembali SPLj, SPAi dan TF masing-masing kegiatan, sehingga akan didapatkan TF masing-masing kegiatan. Untuk lebih detail berapa besarnya TF masing-masing kegiatan dapat di lihat pada tabel berikut ini :

Tabel 5. Kegiatan dengan UREN 45 Hari

Kegiatan	SPL j	Ln	SPA I	TF
A	0	5	0	-5
B	15	5	0	10
C	20	20	5	-5
D	20	5	5	10
E	35	15	25	-5
F	35	15	10	10
G	40	5	10	25
H	40	5	40	-5
I	40	5	25	10
J	45	5	45	-5

Pada tabel di atas dapat dilihat kegiatan-kegiatan yang memiliki total float berharga negatif yaitu kegiatan A-C-E-H-J. Kegiatan-kegiatan tersebut memiliki TF sebesar -5. Langkah berikutnya yang dapat dilakukan adalah dengan menghitung lama kegiatan baru (Ln) menggunakan rumus :

(4)

$$L\,n\,(baru) = Ln\,(lama) + \frac{L\,n\,(lama)}{Li} \times (UREN - UPER)$$

Keterangan :
L n (baru) = Lama kegiatan baru
L n (lama) = Lama kegiatan lama
L i = Jumlah lama kegiatan-kegiatan pada satu lintasan yang harus dipercepat.
UREN = Umur rencana proyek
UPER = Umur perkiraan proyek

Kegiatan-kegiatan yang dapat dipercepat antara lain kegiatan A-C-E-H-J, dikarena kegiatan tersebut memiliki masing-masing TF = UREN-UPER = -5 (berharga negatif). Langkah berikutnya dengan menjumlahkan lama kegiatan-kegiatan pada satu lintasan yang harus dipercepat (Li), maka akan didapatkan :

Li = 5 + 20 + 15 + 5 + 5 = 50

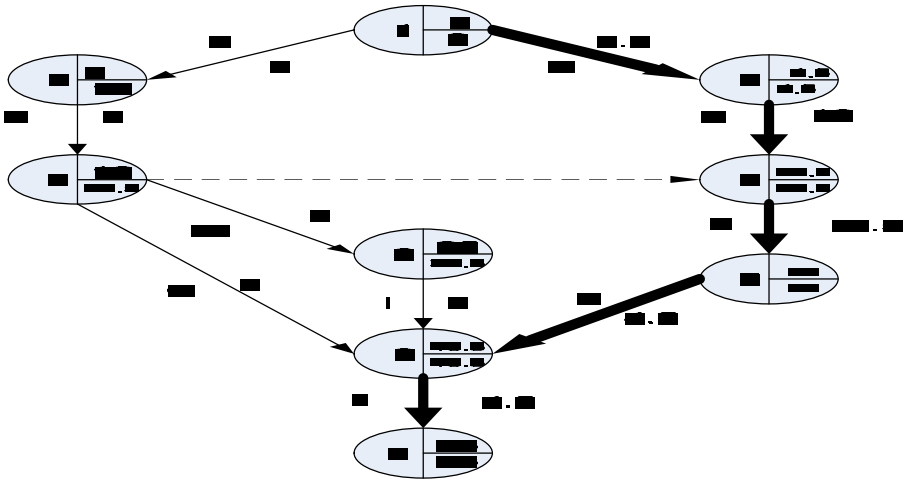
Setelah mendapatkan Li, maka lama kegiatan baru masing-masing kegiatan yang dapat dipercepat akan menghasilkan tabel perkiraan lama kegiatan baru sebagai berikut :

Tabel 6. Kegiatan dengan Lama kegiatan baru

Kegiatan	Lama kegiatan Lama	Lama kegiatan baru = Ln (lama) + Ln (lama) / Li x (UPER - UREN)
A	5	5 + 5 /50 x (-5) = 4.5
C	20	20 + 20 / 50 x (-5) = 18
E	15	15 + 15 / 50 x (-5) = 13.5
H	5	5 + 5 /50 x (-5) = 4.5
J	5	5 + 5 /50 x (-5) = 4.5

Berdasarkan tabel di atas, maka didapatkan lama kegiatan baru masing-masing kegiatan yang dapat dipercepat lama kegiatannya. Langkah berikutnya adalah dengan membuat

kembali diagram PERT/CPM dengan lama kegiatan baru sampai didapatkan TF masing-masing kegiatan bernilai positif. Untuk lebih detailnya dapat di lihat pada gambar berikut ini :



Keterangan :
SPA 1 = 0
SPL 1 = 0
SPA m = 45
SPL n = 45
UPER = UREN = 45

Gambar 7. Diagram PERT/CPM dengan waktu kegiatan baru

Pada gambar di atas dapat di hitung saat paling awal dan saat paling lambat masing-masing kegiatan dengan menggunakan lama kegiatan baru berdasarkan hasil perhitungan lama kegiatan baru, sehingga akan didapatkan UPER =UREN. Setelah UPER dan UREN sama, maka langkah berikutnya adalah dengan menghitung Total Float (TF) masing-masing kegiatan dapat di lihat pada tabel berikut ini

Tabel 7. Kegiatan dengan waktu kegiatan baru

Kegiatan	SPL j	Ln	SPA I	TF
A	4.5	4.5	0	0
B	15	5	0	10
C	22.5	18	4.5	0
D	20.5	5	5	10.5
E	36	13.5	22.5	0
F	35.5	15	10	10.5
G	40.5	5	10	25.5
H	40.5	4.5	36	0
I	40.5	5	25	10.5
J	45	4.5	40.5	0

Berdasarkan tabel di atas dapat di lihat, bahwa Total Fload (TF) masing-masing kegiatan bernilai positif. Berarti proses mempercepat usia proyek telah selesai dilaksanakan dan lama kegiatan baru dapat digunakan untuk menyelesaikan proyek sesuai dengan umur rencana (UREN).

III. PENUTUP

31. Kesimpulan

Dengan menggunakan metode PERT dan CPM kita dapat melakukan percepatan pelaksanaan sebuah proyek dengan menyesuaikan antara umur perkiraan proyek selesai dengan umur rencana proyek yang ditentukan oleh karena suatu hal tertentu.

Berdasarkan PERT dan CPM dapat kita lihat kegiatan-kegiatan mana saja yang dapat dilakukan percepatan dan kegiatan-kegiatan yang masih memiliki toleransi keterlambatan. Apabila sebuah kegiatan mengalami keterlambatan, maka dapat dilihat kegiatan tersebut berada pada lintasan kritis atau tidak. Selama kegiatan tersebut tidak berada di dalam lintasan kritis, maka kegiatan tersebut tidak akan mempengaruhi umur proyek itu sendiri.

Pemanfaatan metode PERT dan CPM dalam mempercepat penyelesaian sebuah proyek pengembangan sistem aplikasi komputer akan dapat diukur tingkat keberhasilannya apabila didukung oleh seluruh komponen yang ada, baik dari pemilik modal, penanggung jawab proyek sampai dengan pelaksana operasional. Waktu yang dibutuhkan oleh masing-masing kegiatan dilaksanakan berdasarkan waktu perkiraan (LPER) yang sudah tercantum di dalam diagram PERT dan CPM dan dilakukan pemantauan terhadap kegiatan-kegiatan kritis yang tidak memiliki waktu tenggang atau toleransi keterlambatan

DAFTAR PUSTAKA

Heizer,Jay.& Barry Render, .2005. Manajemen Operasi.,Jakarta: Salemba Empat.

Indriantoro, Nur dan Supomo Bambang. 2002,. Metodologi Penelitian Bisnis untuk Akuntansi dan Manajemen, Edisi pertama. Yogyakarta: BPFE

Maharesi, Retno. 2002 . Penjadwalan Proyek Dengan Menggabungkan Metode PERT dan CPM. Depok: Universitas Gunadarma. *[Http:/*repository.gunadarma.ac.id:8000/A51-60_Retno_Maharesi_154.pdf, Tgl akses 09/07/2009 pukul 10:52]

Meridith J.R., Mantel SJ, Jr. 2002. *Project Management A Managerial Approach*, Fourth Editions. USA: John Willey & Sons, Inc

Purnama, Rachmat Adi. 1998. Manajemen Proyek. Jakarta:AMIK BSI.

Rostianingsih, Silvia. 2006. Pembuatan Aplikasi Manajemen Proyek Dalam Mengelola Proyek di PT. X.. Depok:Universitas Gunadarma. [http://repository.gunadarma.ac.id:8000/209// Tgl akses 23/07/2009 pukul 09:42]

Schoeder, Roger. 2002. Manajemen Operas., Jakarta: Erlangga.

Soeharto, Iman. 2002. Studi Kelayakan Proyek Industri. Jakarta: Erlangga.

Subagyo, Pangestu. 2000. Dasar-dasar Operations Research. Yogyakarta:BPFE.

Tubagus, Ali Haedar. 1997. Prinsip-prinsip Network Planning. Jakarta:PT Gramedia.Yogyakarta:BPFE.

APLIKASI VISUALISASI MATERI PEMBELAJARAN KIMIA DAN BIOLOGI BERBASIS WEB

Anita Octasia [1)] **Dwiza Riana** [2)]

[1, 2)] Sistem Informasi STMIK Nusa Mandiri
Jl. Kramat Raya No 25 , Jakarta Pusat (10450), Indonesia
anita.octasia@yahoo.com
dwiza_riana@yahoo.com

Abstract
Limitations and difficulties of teachers in delivering materials to make software in educational environments have increased, especially at the senior high school level. That's what makes the writer develops software learning that aims to assist teachers in presenting materials. The materials can help students more easily understand the lessons in school, especially for a chemistry and biology subjects in senior high school. Research methods used in the writing of this research is the analysis of the program. First, the authors analyze the previous program, learn the reference related to the study. They collect the data from books, literature and also from internet browsing. Next, they improve the program by creating new program and proposed in web base aplication. Visualisation and presenting the developed applications uses software such as Adobe Flash CS3 and Macromedia Dreamweaver 8 so that they can create visualized applications of learning materials for biology and chemistry at the high school level. The proposed program is expected to help the teachers in solving the problem of delivering materials.

Keywords
Biologi, Berbasis Web,Kimia, Materi Pembelajaran,

I. PENDAHULUAN

Pada umumnya sistem pendidikan di Indonesia masih menggunakan pendekatan yang berorientasi pada guru, dimana keberhasilan belajar siswa sangat tergantung pada kemampuan dan keterampilan guru. Hal ini tentu saja menyulitkan siswa dalam mempelajari materi pelajaran yang diberikan. Bukan itu saja terkadang media yang digunakan kurang memadai dan menunjang dalam proses pembelajaran.

Karena kesulitan dan keterbatasan para pelaku pendidikan seperti halnya guru dalam penyampaian materi-materi pelajaran, maka kebutuhan akan perangkat lunak pembelajaran di lingkungan sekolah semakin meningkat. Perangkat lunak adalah program komputer yang berfungsi sebagai sarana interaksi antara pengguna dan perangkat keras. Akhir-akhir ini sudah banyak perangkat lunak serta aplikasi multimedia yang dapat memberikan alternatif bagi guru untuk mengatasi kesulitan dalam penyampaian materi yang diajarkannya, tetapi masih ada saja kekurangan.

Internet, suatu istilah yang saat ini sudah tidak asing lagi bagi masyarakat umum. Hal ini bisa kita lihat dari perkembangan pemakaian internet di dunia pada umumnya, dan di Indonesia khususnya. Baik itu dari jumlah komputer pribadi yang terhubung ke internet, komputer jaringan lokal suatu badan/perusahaan yang terhubung ke internet, ataupun jasa warung internet yang menyediakan penyewaan internet untuk umum.

Dengan keadaan seperti itu, internet memegang peranan sangat penting dalam kehidupan manusia dalam berbagai bidang. Seperti bidang informasi, dimana kita dapat mendapatkan informasi terkini dari hasil pertandingan sepak bola dari penjuru dunia, informasi tentang kebudayaan dunia, informasi tentang produk teknologi tercanggih dan terbaru, melakukan transaksi pembelian dan penjualan barang, lelang, konsultasi kesehatan, mendengarkan musik, mencari teman baru, mengobrol lewat internet, dan kegiatan lainnya.

Salah satu sarana internet yang banyak diminati adalah *World Wide Web* (disebut "web") yang mampu menyediakan informasi dalam berbagai media, baik teks, gambar, animasi, maupun kombinasinya. Dengan memanfaatkan kemampuan dan fungsi yang dimiliki oleh *web*, pada tulisan ini penulis hendak membahas mengenai pemanfaatan web sebagai media untuk belajar jarak jauh lewat internet. Informasi yang tersedia dalam web ini dapat diakses oleh siapa saja, dimana saja, dan kapan saja bagi mereka yang ingin belajar. Memang pengembangan web seperti ini sudah dibuat oleh beberapa situs luar negri. Web multimedia yang dibahas pada makalah ini adalah tentang Visualisasi Materi Pembelajaran kimia dan biologi.

Berdasarkan latar belakang di atas, penulis mencoba membuat suatu *website* yang berisi

pemgembangan aplikasi pembelajaran khususnya mata pelajaran kimia dan biologi. Dengan aplikasi ini diharapkan dapat membantu guru dalam menyampaikan materi-materi pelajaran dan dapat membantu siswa lebih memahami dan memgerti materi pelajaran yang dipelajari, khususnya untuk mata pelajaran kimia dan biologi. Karena dengan menggunakan sistem informasi berbasis *web* efektifitas dalam penyampaian informasi dapat dilaksanakan secara cepat dan efisien.

II. PEMBAHASAN

Landasan Teori

A. Visualisasi

Visualisasi adalah konversi data ke dalam format visual atau tabel sehingga karakteristik dari data dan relasi diantara item data atau atribut dapat di analisis atau dilaporkan *(Januar S, 2006)*. Visualisasi data adalah satu dari yang teknik paling baik dan menarik untuk eksplorasi data.

Visualisasi (Inggris: visualization) adalah rekayasa dalam pembuatan gambar, diagram atau animasi untuk penampilan suatu informasi. Secara umum, visualisasi dalam bentuk gambar baik yang bersifat abstrak maupun nyata telah dikenal sejak awal dari peradaban *manusia (J-C J.Jehng, S-H S.Tung, & C-T Chang, 2002).*

Pada saat ini visualisasi telah berkembang dan banyak dipakai untuk keperluan ilmu pengetahuan, rekayasa, visualisasi disain produk, pendidikan, multimedia interaktif, kedokteran, dll. Pemakaian dari grafika komputer merupakan perkembangan penting dalam dunia visualisasi, setelah ditemukannya teknik garis perspektif pada zaman renaissance. Perkembangan bidang animasi juga telah membantu banyak dalam bidang visualisasi yang lebih kompleks dan canggih.

B. Materi Pembelajaran

Materi pembelajaran (*instructional materials*) adalah bahan yang diperlukan untuk pembentukan pengetahuan, keterampilan, dan sikap yang harus dikuasai siswa dalam rangka memenuhi standar kompetensi yang ditetapkan. Materi Pembelajaran menempati posisi yang sangat penting dari keseluruhan kurikulum, yang harus dipersiapkan agar pelaksanaan pembelajaran dapat mencapai sasaran. Materi yang dipilih untuk kegiatan pembelajaran hendaknya materi yang benar-benar menunjang tercapainya standar kompetensi dan kompetensi dasar *(Rusman, 2009).*

Menurut Rusman (2009) dalam panduan pengembangan materi pembelajaran. Jenis-jenis materi pembelajaran dapat diklasifikasi sebagai berikut :

1. **Fakta** yaitu segala hal yang bewujud kenyataan dan kebenaran, meliputi nama-nama objek, peristiwa sejarah, lambang, nama tempat, nama orang, nama bagian atau komponen suatu benda, dan sebagainya. Contoh dalam mata pelajaran Sejarah: Peristiwa sekitar Proklamasi 17 Agustus 1945 dan pembentukan Pemerintahan Indonesia.

2. **Konsep** yaitu segala yang berwujud pengertian-pengertian baru yang bisa timbul sebagai hasil pemikiran, meliputi definisi, pengertian, ciri khusus, hakikat, inti /isi dan sebagainya. Contoh, dalam mata pelajaran Biologi: Hutan hujan tropis di Indonesia sebagai sumber plasma nutfah, Usaha-usaha pelestarian keanekargaman hayati Indonesia secara *in-situ* dan *ex-situ*, dsb.

3. **Prinsip** yaitu berupa hal-hal utama, pokok, dan memiliki posisi terpenting, meliputi dalil, rumus, *adagium*, *postulat*, paradigma, teorema, serta hubungan antarkonsep yang menggambarkan implikasi sebab akibat. Contoh, dalam mata pelajaran Fisika: Hukum Newton tentang gerak, Hukum 1 Newton, Hukum 2 Newton, Hukum 3 Newton, Gesekan Statis dan Gesekan Kinetis, dsb.

4. **Prosedur** merupakan langkah-langkah sistematis atau berurutan dalam mengerjakan suatu aktivitas dan kronologi suatu sistem**.** Contoh, dalam mata pelajaran TIK: Langkah-langkah mengakses internet, trik dan strategi penggunaan *Web Browser* dan *Search Engine*, dsb.

5. **Sikap atau Nilai** merupakan hasil belajar aspek sikap, misalnya nilai kejujuran, kasih sayang, tolong-menolong, semangat dan minat belajar dan bekerja, dsb. Contoh, dalam mata pelajaran geografi: Pemanfaatan lingkungan hidup dan pembangunan berkelanjutan, yaitu pengertian lingkungan, komponen ekosistem, lingkungan hidup sebagai sumberdaya, pembangunan berkelanjutan.

C. Berbasis Web

Web adalah layanan internet yang menggunakan protokol HTTP (*HyperText Transfer Protocol*) disamping layanan *internet* lainnya seperti *Gopher*, *Telnet*, FTP, *E-Mail*, dsb. *Web*

bekerja dengan cara menampilkan *file-file HTML* yang berasal dari *server web* pada program *client* khusus, yaitu *browser web*. Program *browser* pada *client* mengirimkan permintaan (*request*) kepada *server web*, yang kemudian akan dikirimkan oleh *server* dalam bentuk HTML. *File* HTML ini berisi instruksi-instruksi yang diperlukan untuk membentuk tampilan. Perintah HTML ini kemudian diterjemahkan oleh *browser web* sehingga isi informasi dapat ditampilkan secara visual kepada pengguna di layar komputer.

Pengertian aplikasi berbasis *web* dalam tulisan ini adalah aplikasi sisi *server* (*server side*) yang menggunakan standar HTTP dan menggunakan *browser* untuk menggunakan aplikasi. Termasuk didalamnya teknologi CGI, PHP, JSP, ASP dan lainnya. Aplet Java dan teknologi lain seperti Microsoft .NET meskipun menggunakan internet tidak relevan dalam pembahasan ini. Pembahasan dilakukan dari teknologi aplikasi berbasis *web*, otomasi perpustakaan, dan aplikasi otomasi perpustakaan berbasis *web*.

D. Adobe Flash CS3

Sejak diakusisi perusahaan raksasa *Adob*e, maka *software* multimedia *Macromedia Flash* berubah nama menjadi *Adobe Flash*. Akusisi ini pun bisa jadi merupakan pertanda merupakan bahwa prospek pembuatan animasi *flash* akan semakin berkembang.

Flash sudah dipakai sejak puluhan tahun yang lalu. Sebagian kalangan menggunakan untuk membuat animasi yang halaman *websit*e, profil perusahaan, cd interaktif, *game* di *mobile device* seperti di handphone dan PDA.

Setiap *software* memiliki kelebihan dan kekurangan. *Adobe Photoshop* memiliki fitur untuk menggambar yang luar biasa, tetapi tidak bisa menganimasikan. *Adobe After Effect* memiliki kemampuan animasi yang luar biasa, tapi tidak untuk menggambarkan objek. Objek-objek yang digunakan dalam *Adobe After Effe*ct adalah import dari *output software* lain. *Software 3D Studio Max* jau lebih dasyat, bisa menggambarkan objek 3 dimensi dan menganimasikannya. Namun, perlu tenaga ekstra untuk mempelajarinyan karena terlalu banyak fiturnya.Selain memiliki kemampuan untuk menggambar, flash juga bisa sekaligus menganimasikannya. Memang efek-efek gambarnya tidak secanggih dan seberagam *adobe photosop*, tapi sudah cukup untuk menggambarkan objek-objek agar terlihat cantik dan artistik.

Didalam flash, kita bisa memasukkan rumus fisika, matematik atau rumus-rumus lainnya dalam bentuk *action script*. Sehingga kita bisa mensimulasikan mobil yang bergerak dengan kecepatan dan percepatan tertentu. Semuanya menjadi mungkin dan mudah dengan *flash*.

Metode Penelitian

Dalam rangka pengumpulan data yang diperlukan untuk bahan penulisan ini, penulis menggunakan beberapa metode penelitian, yaitu :

1. **Analisa Program**
   Untuk memahami program aplikasi animasi pendidikan yang akan dikembangkan, menganalisa kekurangan yang ada dan kemudian mengembangkan program tersebut untuk mengurangi kekurangan program tersebut.

2. **Studi Literatur**
   Metode pengumpulan data yang bersumber dari buku-buku terutama yang berkaitan dengan nateri-materi pelajaran di bidang kimia dan biologi.

3. **Pengembangan Program**
   Pengembangan program dapat diartikan sebagai kegiatan membuat program baru untuk menggantikan fungsi program yang ada. Dengan membuat tampilan yang interaktif dan dinamis pada program tersebut, dapat diakses oleh siapa saja dan kapan saja dan juga program tersebut dapat di u*pdate* di waktu-waktu tertentu.

TOOLS SYSTEM

A. *Entity Relationship Diagram* (ERD)

Entity Relationship Diagram (ERD) adalah suatu penyajian data dengan menggunakan *entity* dan *relationship* yang digunakan untuk menganalisa data dan memecahkan masalah. ERD (*Entity Relationship Diagram*) dapat digunakan untuk menganalisa masalah dan untuk memecahkan masalah (Primasanti ,2007).

Menurut Primasanti (2007) pembuatan ERD dapat dilakukan dengan langkah sebagai berikut:

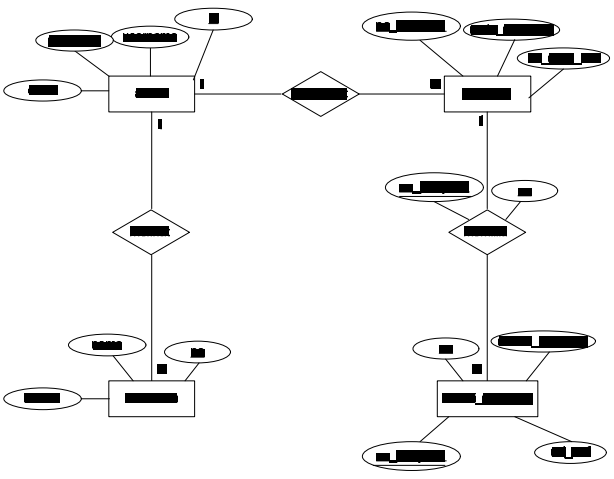
1. Keluarkan semua atribut yang dimiliki oleh dokumen sumber.
2. Tentukan antribut yang ada menjadi *primary key*, jika tidak ada boleh dibuat baru lalu tentukan ketergantugan atribut terhadap *primary key*-nya.

3. Tentukan nama entity dari kelompok dari kelompok atribut yang telah bergantung terhadap *primary key*-nya.
4. Gambarkan hubungan masing-masing *entity* beserta atributnya.
5. Tentukan *cardinality* atau tingkat hubungan dari masing-masing *entity* yang telah terhubung.

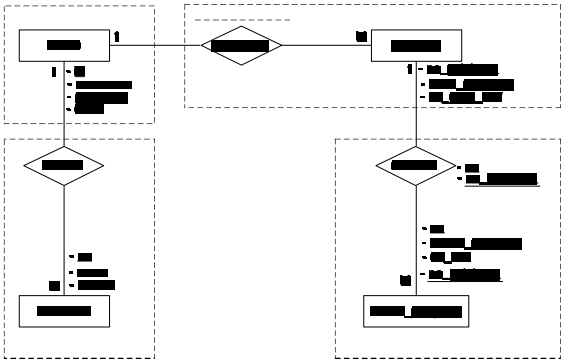
B. Transformasi ERD ke *Logical Record Structure* (LRS)

Berikut adalah cara bagaimana mentransformasikan ERD menjadi LRS dimana setiap *entity* akan dirubah kedalam bentuk kotak dengan nama entity berada di luar dan atribut berada did alam kotak tersebut. Dalam transformasi ERD ke LRS, *cardinality* sangat berpengaruh dalam penggabungan antara *entity* dan *relationship* untuk pembentukan suatu LRS.

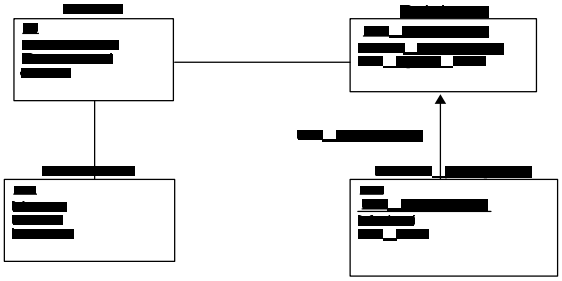
1. Penggabungan *entity* dan *relationship* yang memiliki *cardinality* 1:1, *relationship* digabungkan pada *entity* terakhir, sehingga membentuk dua buah LRS
2. Penggabungan *entity* dan *relationship* yang memiliki *cardinality* 1:M atau M:1, *relationship* digabungkan pada *entity* yang memiliki *cardinality* M (*many*), sehingga membentuk dua buah LRS.
3. Penggabungan *entity* dan *relationship* yang memiliki *cardinality* M:N, *relationship* tidak digabungkan pada *entity*. Sehingga dari hasil ini akan terbentuk tiga buah LRS.



Gambar 1. ER-Diagram



Gambar 2. Transformasi ERD ke LRS



Gambar 3. LRS (*Logical Record Structure*)

C. *Unified Modelling Language* (UML)

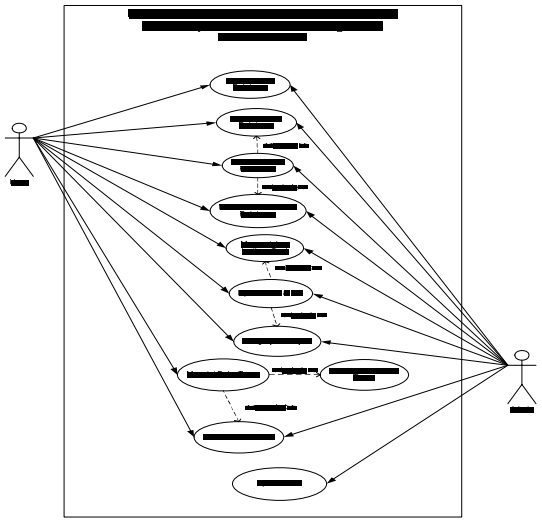
Unified Modelling Language (UML) adalah salah satu alat bantu yang sangat handal di dunia pengembangan sistem yang berorientasi obyek. UML merupakan kesatuan dari bahasa pemodelan yang dikembangkan oleh Booch, *Object Modelling Technique* (OMT) dan *Object Oriented Software Engineering* (OOSE). Metode Booch sangat terkenal dengan nama metode *Design Object Oriented*

Keunggulan metode Booch adalah pada detil dan kayanya dengan notasi dan elemennya. Pemodelan OMT yang dikembangkan oleh Rumbaugh didasarkan pada analisa terstruktur dan pemodelan *entity-relationship*. Dengan UML, metode Booch, OMT dan OOSE digabungkan dengan membuang elemen-elemen yang tidak praktis ditambah dengan elemen-elemen dari metode lain yang lebih efektif dan elemen-elemen baru yang belum ada pada metode terdahulu. Sehingga UML lebih ekspresif dan seragam daripada metode lainnya. Dan dengan UML kita dapat membuat model untuk semua jenis aplikasi piranti lunak yang dapat berjalan pada piranti keras, sistem operasi dan jaringan apapun, serta ditulis dalam bahasa

pemrograman apapun. Yang terdiri dari beberapa diagram yaitu :

1. Use Case Diagram

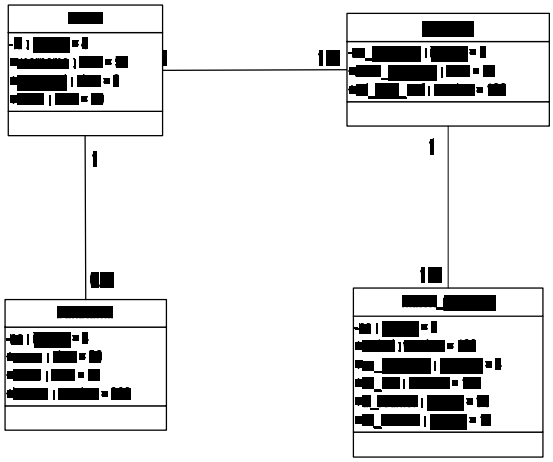
Use case diagram menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Yang ditekankan adalah "apa" yang diperbuat sistem, dan bukan "bagaimana". Sebuah *use case* mempresentasikan sebuah interaksi antara aktor dengan sistem. *Use case* merupakan sebuah pekerjaan tertentu, misalnya login ke sistem, meng-*create* sebuah daftar belanja, dan sebagainya. Seorang atau sebuah aktor adalah sebuah entitas manusia atau mesin yang berinteraksi dengan sistem untuk melakukan pekerjaan-pekerjaan tertentu. *Use case diagram* dapat sangat membantu bila kita sedang menyusun *requirement* sebuah sistem, mengkomunikasikan rancangan dengan klien, dan merancang *test case* untuk semua *feature* yang ada pada sistem. Sebuah *use case* dapat meng-*include* fungsionalitas *use case* lain sebagai bagian dari proses dalam dirinya. Secara umum diasumsikan bahwa *use case* yang di-*include* akan dipanggil setiap kali *use case* yang meng-*include* dieksekusi secara normal. Sebuah *use case* dapat di-*include* oleh lebih dari satu *use case* lain, sehingga duplikasi fungsionalitas dapat dihindari dengan cara menarik keluar fungsionalitas yang *common*. Sebuah *use case* juga dapat meng-*extend use case* lain dengan *behaviour*-nya sendiri. Sementara hubungan generalisasi antar *use case* menunjukkan bahwa *use case* yang satu merupakan spesialisasi dari yang lain.



Gambar 4. *Usecase Diagram*

2. Class Diagram

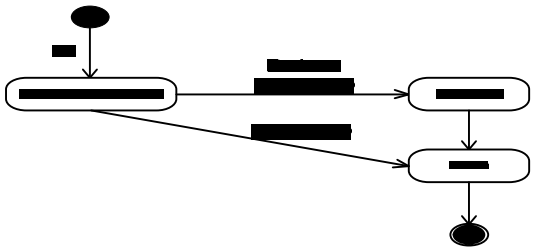
Class adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek *(Munawar, 2005).*



Gambar 5. *Class Diagram*

3. Statechart Diagram

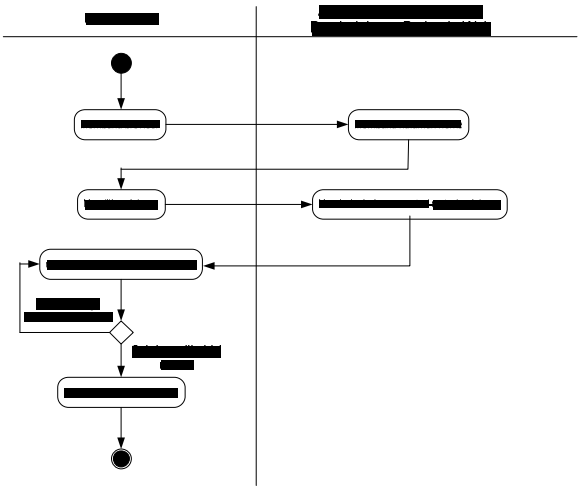
Statechart diagram menyediakan variasi simbol dan sejumlah ide untuk pemodelan. Tipe diagram ini mempunyai potensi untuk menjadi sangat kompleks dalam waktu yang singkat. *Statechart diagram* digunakan untuk menggambarkan bagaimana event mengubah sebuah obyek. Pada aplikasi ni terdapat beberapa *Statechart diagram* sesuai dengan kebutuhan aplikasi, yang terdiri dari : *Statechart diagram* buku tamu, *Statechart diagram* input data admin, *Statechart diagram* pelajaran, dan *Statechart diagram* materi pelajaran, dengan bentuk diagram yang sama baik informasi yang mengalir antar state maupun aliran informasi yang mengalir antar state sesuai dengan kebutuhan aplikasi.



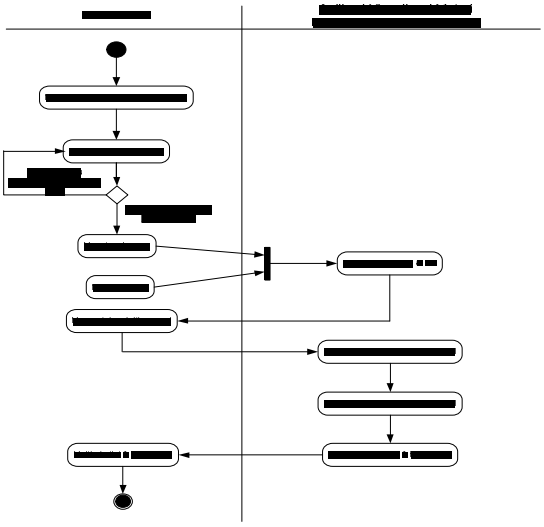
Gambar 6. ***Statechart Diagram***

4. ***Activity Diagram***

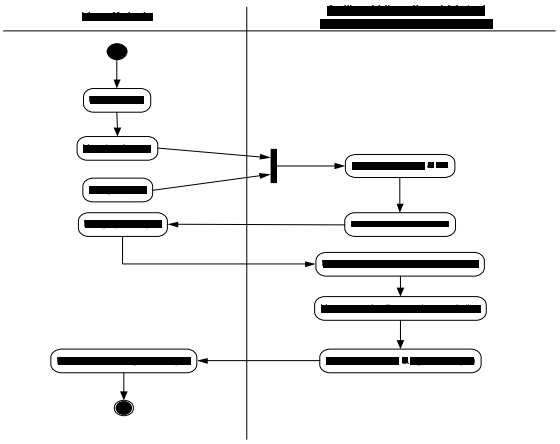
Activity diagram adalah teknik untuk mendiskripsikan logika prosedural, proses bisnis dan aliran kerja dalam banyak kasus. Activity diagram mempunyai peran seperti halnya *flowchart*, akan tetapi perbedaannya dengan flowchart adalah activity diagram bisa mendukung perilaku paralel sedangkan flowchart tidak bisa *(Munawar, 2005)*. Pada aplikasi ni terdapat beberapa *Activity diagram* sesuai dengan kebutuhan aplikasi, yang terdiri dari :



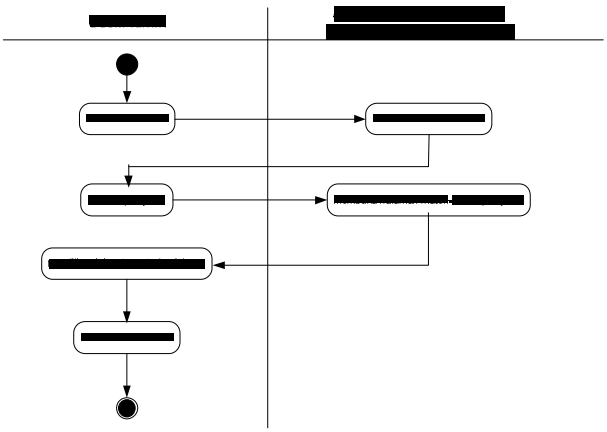
Gambar 7. ***Activity Diagram*** **Melihat Materi-Materi Pelajaran**



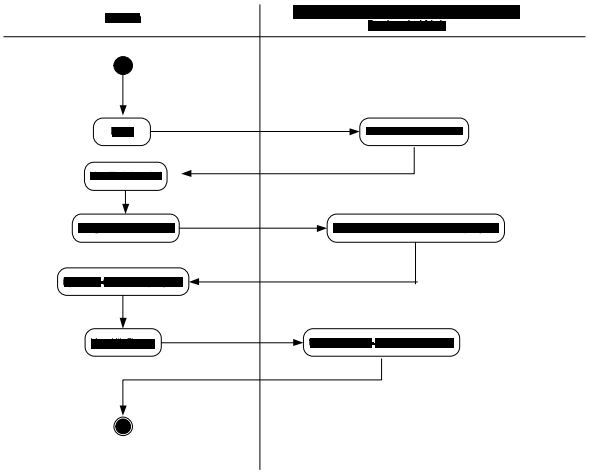
Gambar 8. ***Activity Diagram*** **Mengerjakan Latihan Soal**



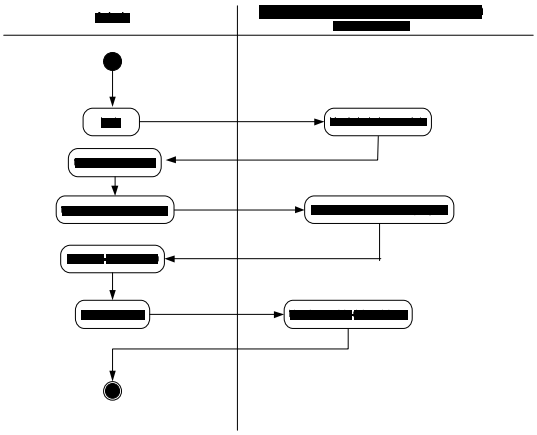
Gambar 9. ***Activity Diagram*** **Mengerjakan Ujian**



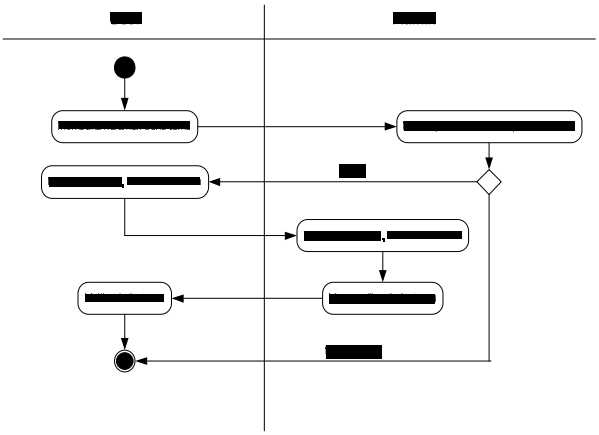
Gambar 10. ***Activity Diagram*** **Men-*download* Materi**



Gambar 12. ***Activity Diagram*** **Menambah Materi Pelajaran**



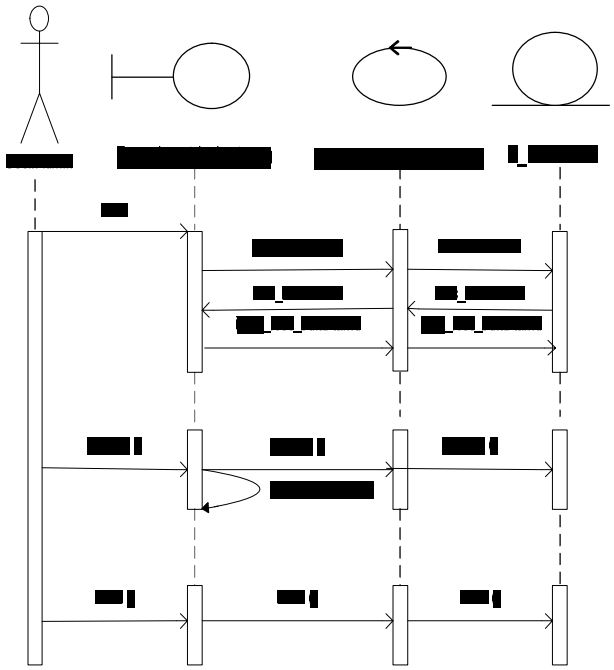
Gambar 11. ***Activity Diagram*** **Menambah Pelajaran**



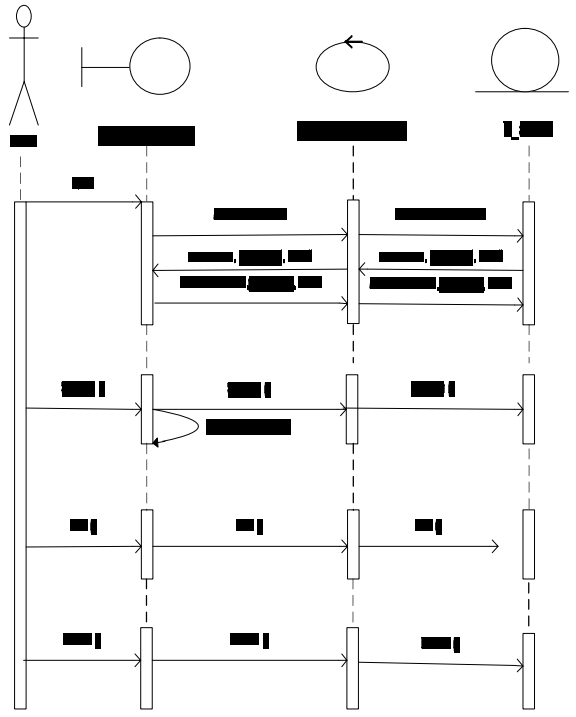
Gambar 13. ***Activity Diagram*** **Melihat Buku Tamu**

5. Sequence Diagram

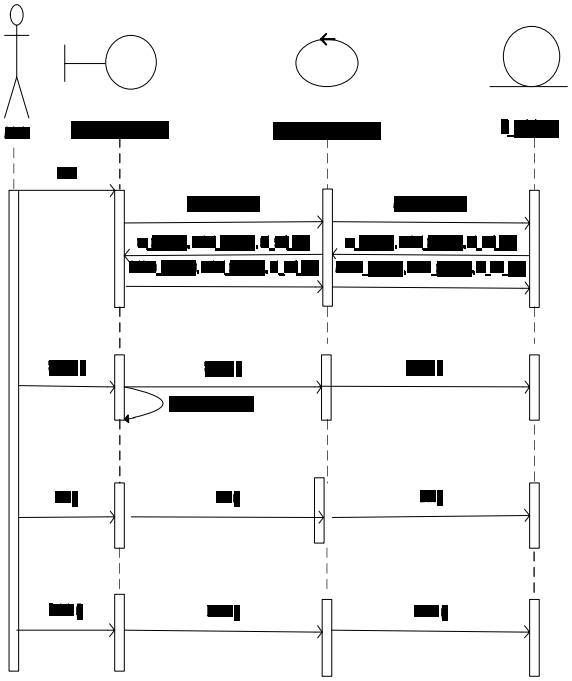
Sequence diagram digunakan untuk menggambarkan perilaku pada sebuah *scenario*. Diagram ini menunjukkan sejumlah contoh objek dan *message* (pesan) yang diletakkan diantara objek-objek ini di dalam *use case. Sequence diagram* digunakan untuk menggambarkan perilaku pada suatu urutan kejadian. Pada aplikasi ni terdapat beberapa *Activity diagram* sesuai dengan kebutuhan aplikasi, yang terdiri dari :



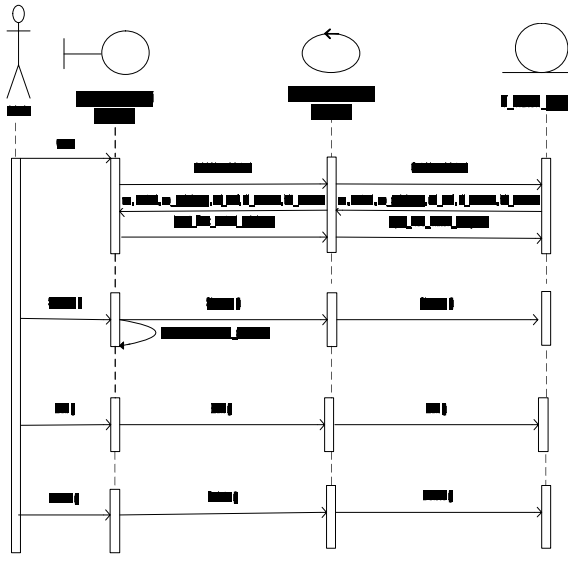
Gambar 14. *Sequence Diagram* Buku Tamu



Gambar 15. *Sequence Diagram* Input Data Admin



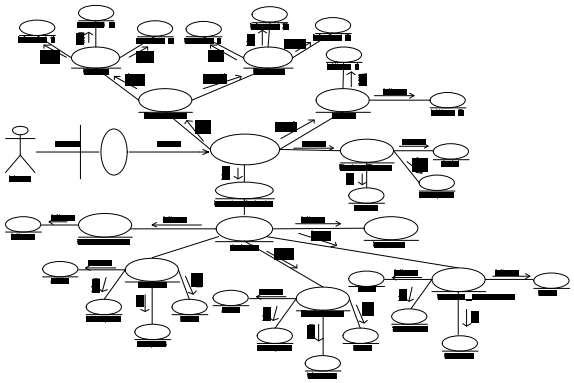
Gambar 16. *Sequence Diagram* Input Data Pelajaran



Gambar 17. *Sequence Diagram* Input Data Materi Pelajaran

6. *Collaboration Diagram*

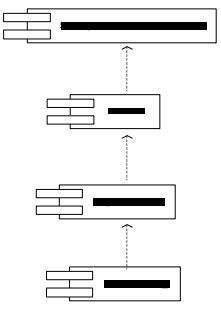
Collaboration diagram juga menggambarkan interaksi antar objek seperti *sequence diagram*, tetapi lebih menekankan pada peran masing-masing objek dan bukan pada waktu penyampaian *message. Collaboration diagram* diorganisir menurut ruang dan digunakan untuk menunjukkan message-message obyek yang dikirimkan satu sama lain.



Gambar 18. *Collaboration Diagram*

7. *Component Diagram*

Component diagram adalah implementasi *software* dari sebuah *class. Class* mewakili abstraksi dari serangkaian *attribute* dan *operation (Munawar, 2005).* Hal terpenting yang perlu diingat tentang *class* dan *component* adalah sebuah *component* bisa jadi merupakan implementasi dari lebih dari sebuah *class* .



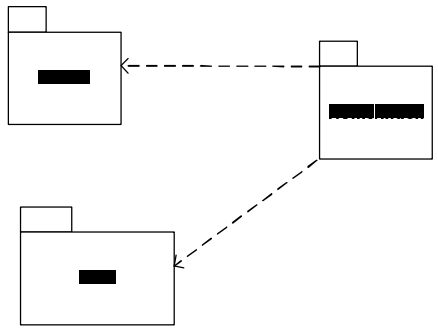
Gambar 19. *Component Diagram*

8. *Deployment Diagram*

Deployment/physical diagram menggambarkan detail bagaimana komponen di-*deploy* dalam infrastruktur sistem, di mana komponen akan terletak (pada mesin, *server* atau piranti keras apa), bagaimana kemampuan jaringan pada lokasi tersebut, spesifikasi *server*, dan hal-hal lain yang bersifat fisikal.

9. *Package Diagram*

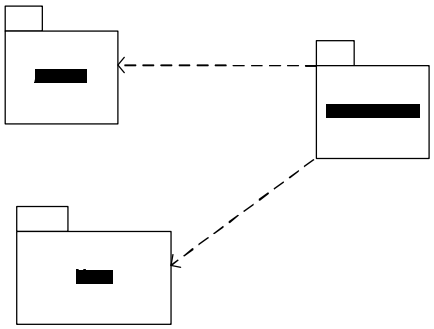
*Package Diagram a*dalah sebuah bentuk pengelompokkan yang memungkinkan untuk mengambil sebuah bentuk di UML dan mengelompokkan elemen – elemennya dalam tingkatan unit yang lebih tinggi *(Munawar,2005).* Kemampuan *package* yang paling umum adalah untuk mengelompokkan *class.* Pada *package diagram* aplikasi visualisasi pembelajaran berbasis *web* dan dapat digunakan untuk penggambaran saling ketergantungan antara *component-component* utama dengan sistem.



Gambar 21. *Package Diagram*

9. *Package Diagram*

*Package Diagram a*dalah sebuah bentuk pengelompokkan yang memungkinkan untuk mengambil sebuah bentuk di UML dan mengelompokkan elemen – elemennya dalam tingkatan unit yang lebih tinggi *(Munawar, 2005).* Kemampuan *package* yang paling umum adalah untuk mengelompokkan *class.* Pada *package diagram* aplikasi visualisasi pembelajaran berbasis *web* dan dapat digunakan untuk penggambaran saling ketergantungan antara *component-component* utama dengan sistem.



Gambar 21. *Package Diagram*

V. WEB BASE SEBAGAI MEDIA VISUALISASI MATERI PEMBELAJARAN

Dilihat dari kebutuhan aplikasi visualisasi materi pembelajaran kimia danbBiologi berbasis *web*, aplikasi ini mampu untuk menunjang kegiatan para guru dalam mengajar, dapat memotivasi siswa untuk belajar dan dapat membantu siswa dalam memahami materi pelajaran yang disampaikan. Aplikasi ini adalah aplikasi *open source* dan dapat dgunakan secara gratis, sehingga dapat di akses oleh siapa saja. Berikut tampilan aplikasi visualisasi materi pembelajaran berbasis *web*, yang terdiri dari:

1. Halaman Home (Halaman Depan Web)

Gambar 22. Halaman Home

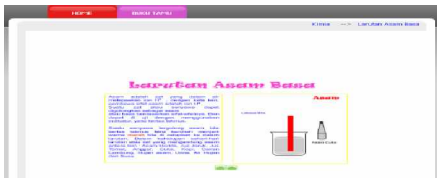
2. Halaman Pelajaran

Berikut adalah salah satu halaman yang berisi materi-materi yang ingin disajikan dan dapat didownload oleh user.

Gambar 23. Halaman Pelajaran

3. Halaman Materi Pelajaran

Berikut adalah salah satu halaman materi pelajaran yang berisi penyajian salah satu materi materi pelajaran yang ingin disajikan.



Gambar 24. Halaman Materi Pelajaran

4. Halaman Ujian

Berikut salah satu halaman ujian yang berisi soal-soal dari keseluruhan materi pelajaran yang telah disajikan, untuk menguji apakah siswa memahami materi yg disampaikan atau tidak.

Gambar 25. Halaman Ujian

5. Halaman Buku Tamu

Halaman ini berfungsi sebagai halaman yang menampilkan komentar atau kritik dan saran yang membangun.

Gambar 26. Halaman Buku Tamu

Dalam aplikasi visualisasi materi pembelajaran berbasis web ini, pelajaran dan materi pelajaran dapat ditambah oleh seorang admin, sesuai dengan kebutuhan. Dimana seorang admin mempunyai ruang tersendiri dalam melakukan pekerjaannya. Berikut adalah tampilan ruang admin, yang terdiri dari :

1 . Halaman Login Admin

Halaman ini berfungsi agar admin dapat masuk diruang admin.

Gambar 27. Halaman Login Admin

2 Halaman Home Admin

Halaman ini hanya dapat diakses oleh seorang admin, ruang ini terdapat fasilitas untuk mengapdate data admin, mata pelajaran dan materi-materi pelajaran.

Gambar 28. Halaman Home Admin

3 Halaman Admin

Halaman ini berfungsi untuk mengupdate data admin dan halaman ini hanya dapat diakses oleh seorang admin.

Gambar 29. Halaman Admin

4 Halaman Pelajaran

Halaman ini berfungsi untuk mengupdate data mata pelajaran dan halaman ini hanya dapat diakses oleh seorang admin.

Gambar 29. Halaman Pelajaran

5 Halaman Materi-Materi

Halaman ini berfungsi untuk mengupdate materi-materi pelajaran dan halaman ini hanya dapat diakses oleh seorang admin.

Gambar 30. Halaman Materi-Materi

III. PENUTUP

3.1. Kesimpulan

Dalam menganalisa dan merancang suatu program tentunya dibutuhkan kreatifitas, ketekunan dan tingkat ketelitian yang tinggi agar animasi yang dibuat dapat diterima oleh pengguna. Beberapa kesimpulan yang dapat diambil sebagai berikut:

1. Terdapat kesulitan para pengajar untuk dapat menyampaikan materi-materi dengan tidak hanya menggunakan kata-kata tetapi dapat ditampilkan secara visual dalam bentuk animasi.
2. Masih sulitnya membuat aplikasi yang dapat menunjang materi-materi pelajaran di sekolah khususnya pelajaran kimia dan biologi.
3. Untuk mengatasi kesulitan dan masalah tersebut metode yang digunakan adalah dengan melakukan perancangan suatu aplikasi dengan menggunakan web base sebagai media visualisasi materi pembelajaran.
4. Hasil dari aplikasi yang dibuat dapat membantu pengajar dalam menyampaikan materi dan dapat menunjang materi-materi

pelajaran khususnya mata pelajaran kimia dan biologi. Materi pelajaran yang disajikan dalam bentuk visualisasi dapat memotivasi mahasiswa untuk belajar dengan mudah dan menyenangkan.Dengan tampilan yang menarik dan mudah untuk dioperasikan.

DAFTAR PUSTAKA

Efendy, Rusman. Panduan Pengembangan Materi Pembelajaran Depdiknas.http://www.dikmenum.go.id/dataapp/kurikulum/ [diakses tanggal 18 Oktober 2009].

Hidayatullah, Priyanto, M. Amarullah Akbar, dan Zaky Rahim. *Making Educational Animation Using Flash*. Bandung: Informatika.(2008).

J. Jehng, J-C, S. Tung, S-H & Chang, C-T. *A visualization Approach to Learnig The Concept of Recursion*. *Journal of Computer assited Learning 15,279-290*. (2002).

Munawar. Pemodelan Visual dengan UML. Graha Ilmu. Yogyakarta. (2005).

Primasanti, Ayu Y Ida. *Entity Relation Diagram*. (2007).[http://*iaprima.staff.gunadarma.ac.id/Downloads/files/5459/Bahasan9a_ERD.pdf*} (Akses 25 Agustus 2009).

S, Januar. 2006. Pengenalan Teknologi Komputer dan Informasi. Yogyakarta. Andi Offset.

INDEKS SUBJEK

Vulnerabilities, 110,114,115,121,135
Packet Filtering Firewall, 110,111,117,120
Cisco IP Access Control List (ACL), 110,112,119
Cracker, 110,136
Wireless, 122,126,127,128
Network, 110,111,117,120,121,122,123,125,129,131,132,133,134,135,136,137,138,142,143,146,156,157,174,175,176,177,178,179,180,181,186,188,201
internet protocol, 122,125,129
hacker, 134,135,141,142,143
Intrusion Detection System, 120,134,136,137,138,142
snort for windows, 134
E-mail, 134,142,144,145,146,147,148,149,150,153,154,155,156
Spam, 144,145,146,147,148,149,153,154,155,156
E-mail filter, 144
SMTPI Methodics, 144
Web-Based Employee Information System, 157
Technology Accepted Model (TAM), 157,172
Structural Equations Modelling (SEM), 157
Analysis of MOment Structure (AMOS), 157
PERT, 174,175,176,180,182,184,186
CPM, 174,175,176,182,186,187
UPER, 174,175,180,185,186,187
UREN, 174,180,181,182,184,185,186,187,188
Biologi, 115,189,190,191,198,200
Berbasis Web, 138,157,158,160,161,162,167,168,170,171,172,189,190,198,199
Kimia, 189,190,191,198,200
Materi Pembelajaran, 189,190,198,199,200

INDEKS PENULIS

PARADIGMA, Vol. XI, No. 1 Maret 2009

Hendra Supendar, RANCANG BANGUN SISTEM PENERIMAAN PEGAWAI PADA KANTOR PUSAT PERUSAHAAN UMUM (PERUM) PEGADAIAN JAKARTA. Vol. XI, No. 1 Maret 2009. Halaman 1 – 11.

Ishaq Kholil, KAJIAN PENERAPAN SISTEM KOMPUTERISASI KENAIKAN PANGKAT REGULER PADA PUSAT DATA DAN INFORMASI KETRANSMIGRASIAN DEPARTEMEN TENAGA KERJA DAN TRANSMIGRASI JAKARTA. Vol. XI, No. 1 Maret 2009. Halaman 12 - 23

Anton, ANALISA JARINGAN LOCAL AREA NETWORK PADA AKADEMI BINA SARANA INFORMATIKA KAMPUS FATMAWATI A1 JAKARTA. Vol. XI, No. 1 Maret 2009. Halaman 24 – 34.

Dedi Saputra, PEMBUATAN ALAT PENGHITUG BANYAKNYA ORANG YANG MASUK RUANGAN DENGAN SENSOR INFRA MERAH MENGGUNAKAN APLIKASI VISUAL BASIC 6.0. Vol. XI, No. 1 Maret 2009. Halaman 35 – 47.

Suryanto, ANALISA JARINGAN KOMPUTER PADA PUSAT DATA DAN INFORMASI SEKRETARIAT JENDERAL DEPARTEMEN PERHUBUNGAN RI JAKARTA PUSAT. Vol. XI, No. 1 Maret 2009. Halaman 48 - 57.

PARADIGMA, Vol. XI, No. 2 September 2009

Imam Sutoyo dan Mochamad Wahyudi. KAJIAN PENGGUNAAN PACKET FILTERING FIREWALL MENGGUNAKAN CISCO IP ACCESS CONTROL LIST. Vol. XI, No. 2 September 2009. Halaman 110 – 121.

Agus Dendi Rachmatsyah. INSTALLASI DESAIN DAN PENGEMBANGAN JARINGAN KOMPUTER BERSKALA KECIL(STUDI KASUS : INSTALASI *ONLINE WIRELESS* DI RUMAH). Vol. XI, No. 2 September 2009. Halaman 122 – 133

Hendra Supendar dan Tunggul Yogi Hernowo. PENERAPAN *INTRUSION DETECTION SYSTEM* SEBAGAI FIREWALL DAN SARANA UNTUK MENANGKAL PENYUSUPAN *HACKER* PADA JARINGAN LOKAL PT. ASURANSI JIWA INHEAL INDONESIA, DI JAKARTA . Vol. XI, No. 2 September 2009. Halaman 134 – 143.

Nandang Iriadi. MODEL PENANGANAN *E-MAIL SPAM* DENGAN MENGGUNAKAN METODE SMTPI. Vol. XI, No. 2 September 2009. Halaman 144– 156.

Mochamad Wahyudi. KAJIAN PENERAPAN SISTEM INFORMASI KARYAWAN BERBASIS *WEB* BERDASARKAN PENDEKATAN TAM. Vol. XI, No. 2 September 2009. Halaman 157 – 172

Rachmat Adi Purnama. MEMPERCEPAT USIA PROYEK MENGGUNAKAN METODE PERT DAN CPM UNTUK PENGEMBANGAN SISTEM APLIKASI KOMPUTER. Vol. XI, No. 2 September 2009 Halaman 173 – 188.

Anita Octasia dan Dwiza Riana. APLIKASI VISUALISASI MATERI PEMBELAJARAN KIMIA DAN BIOLOGI BERBASIS WEB. Vol. XI, No. 2 September 2009. Halaman 189 – 196.

PEDOMAN PENULISAN JURNAL ILMIAH PARADIGMA

1. Naskah adalah asli, belum pernah diterbitkan/dipublikasikan di media cetak lain dan ditulis dengan ragam Bahasa Indonesia baku atau dalam Bahasa Inggris.
2. Naskah yang dimuat dalam Jurnal meliputi tulisan tentang gagasan konseptual, kajian dan aplikasi teori, studi kepustakaan dan hasil penelitian. Tulisan Fokus pada bidang Sain, Manajemen Informatika dan Komputer.
3. Isi naskah terdiri dari (a) Judul, (b) Nama Penulis; tanpa gelar, (c) Abstrak, (d) Pendahuluan, (e)Tinjauan Pustaka (f) Metode Penelitian (g) Hasil dan Pembahasan (h) Kesimpulan, (i) Saran, (j) Daftar Pustaka.
4. Naskah diketik dalam 1 (satu) spasi dengan menggunakan Ms. Word (Font Times New Roman, ukuran 10 pitch), dengan jumlah kata minimal 3500 kata atau 9 – 12 halaman kertas A4 (sudah termasuk gambar, table, ilustrasi, dan daftar pustaka), dengan batas pengetikan adalah batas kiri = 4 cm, batas kanan, batas atas = 3 cm, dan batas bawah = 2,5 cm.
5. Judul tidak boleh lebih dari 14 kata dalam tulisan Bahasa Indonesia atau 10 kata dalam Bahasa Inggris. Abstrak berisi tidak lebih dari 250 kata dan merupakan intisari seluruh tulisan yang meliputi: latar belakang, tujuan, metode, hasil dan kesimpulan serta ditulis dalam Bahasa Inggris cetak miring. Diketik 1 spasi. Di bawah abstrak disertakan 3-5 kata kunci (*key word*).
6. Daftar Pustaka berisi informasi tentang sumber pustaka yang dirujuk dalam tubuh tulisan. Format perujukan pustaka mengikuti Sistem *Harvard*.

Penulisan Daftar Pustaka Sistem Harvard (author-date style)
Sistem Harvard menggunakan nama penulis dan tahun publikasi dengan urutan pemunculan berdasarkan nama penulis secara alfabetis. Alamat Internet ditulis cetak miring.

Contoh :

Buller H, Hoggart K. 1994a. *New drugs for acute respiratory distress syndrome*. New England J Med 337(6): 435-439.

Buller H, Hoggart K. 1994b. *The social integration of British home owners into French rural communities.* J Rural Studies 10(2):197–210.

Dower M. 1977. *Planning aspects of second homes.* Di dalam Coppock JT (ed.), *Second Homes: Curse or Blessing*? Oxford: Pergamon Pr. Hlm 210–237.

Grinspoon L, Bakalar JB. 1993. *Marijuana: the Forbidden Medicine*. London: Yale Univ Pr.

Skjellum, Anthony, Gregory Henley, Nathan Doss, and Thomas McMahon. *A guide to writing Myrinet control programs for LANai 3.x. Tutorial Myrinetcontrolprograms* [*http://www.erc.msstate.edu/labs/icdcrl/ learn_mcp/smp.ps]* (Accessed 8 Agustus 2003).

7. Naskah diserahkan kepada LPPM BSI dan Nusa Mandiri , berupa disket/CD data dan print-out (cetakan) dari Tulisan Ilmiah yang dibuat, atau di kirim lewat e-mail ke jurnalparadigma@yahoo.co.id .
8. Isi tulisan bukan merupakan tanggung jawab redaksi. Redaksi berhak mengedit redaksional tanpa mengubah arti.
9. Redaksi berhak menolak naskah yang tidak memenuhi syarat dan akan dikembalikan.
10. Hal-hal yang belum jelas dapat menghubungi LPPM BSI dengan alamat:
    Jl Dewi Sartika No. 77 Cawang, Jakarta Timur 13630. Telp: 021-8000063, Ext: 232.
    Email :lppmbsi@yahoo.com, lppmbsi@bsi.ac.id

RIWAYAT HIDUP PENULIS

Agus Dendi Rachmatsyah, bertempat dan tanggal lahir di Mentok, 31 Agustus 1979, bekerja sebagai staf akademik di Bina Sarana Informatika. Riwayat pendidikan beliau adalah S1 Teknik Informatika. Sedangkan hobi beliau adalah mendengarkan musik, membaca dengan alamat email agus_dnd@yahoo.com.

Dwiza Riana, bertempat dan tanggal lahir di Palembang, 22 Oktober 1970, saat ini beliau menjabat sebagai ketua STMIK Nusa Mandiri Jakarta dan memiliki jabatan fungsional akademik Lektor Kepala. Adapun penulisan jurnal yang telah dihasilkan antara lain, Sistem Informasi Eksekutif Pemasaran Pada PT Datapati Tara Andhika Pilar Nusa Mandiri Journal of Computing and Information (Vol. III No.5 September 2008), Pemodelan Nilai Persentil Data Antropometri Siswa Guna Perancangan Fisik Kursi Dan Meja Sekolah Dengan Perangkat Lunak Humancad Mannequin. Techno Journal of Computing and Information Technology ISSN:1978-2136 Vol IV No.6. Maret 2008, Program Menghitung Bangun Datar dan Bangun Ruang untuk Siswa Kelas 5 dan 6 SD Menggunakan Microsoft Visual Basic Pilar Nusa Mandiri Journal of Computing and Information (Vol. IV. No.6, Maret 2008), Technology dalam dua tahun terakhir, seperti Proxy Server sebagai Web Filtering. Pilar Nusa Mandiri Journal of Computing and Information (Vol.III No.4 Maret 2007), dan Tingkat kepentingan dan kepuasan Mahasiswa Terhadap KinerjaBagian Administrasi pada Bina Sarana Informatika. Jurnal Cakrawala (ISSN 1411 -8629) Vol.5. No.1/2005. Itulah sekelumit penulisan jurnal yang telah beliau lakukan disamping masih banyak lagi penulisan jurnal dan artikel serta seminar yang telah dihasilkan oleh beliau dengan alamat email **dwiza_riana@yahoo.com**.

Hendra Supendar, Pendidikan terakhir Strata Satu (S1) di Universitas Persada Indonesia "YAI" dengan konsentrasi pada jurusan Teknik Informatika, lulus pada tahun 1997/1998. Menjadi Staff Pengajar sejak tahun 1999 hingga saat ini, di lembaga pendidikan Bina Sarana Informatika (BSI), pada program studi Manajemen Informatika, Teknik Komputer dan Akuntansi Komputer untuk matakuliah yang berkaitan dengan komputer, pada bidang *software* dan *hardware*.

Imam Sutoyo, lahir di Jakarta pada tanggal 17 Februari 1982. Pendidikan terakhir Strata Satu (S1) Universitas Gunadarma, Fakultas Ilmu Komputer, Jurusan Sistem Komputer, lulus pada tahun 2005. Saat ini bekerja sebagai staff akademik di Bina Sarana Informatika kampus Jatiwaringin.

Nandang Iriadi, Pendidikan terakhir Strata Satu (S1) di Universitas Respati Indonesia dengan konsentrasi pada jurusan Teknik Informatika, lulus pada tahun 2002/2003. Menjadi Staff Pengajar sejak tahun 2004 hingga saat ini, di lembaga pendidikan Bina Sarana Informatika (BSI), pada program studi Manajemen Informatika, Teknik Komputer dan Akuntansi Komputer untuk matakuliah yang berkaitan dengan komputer, pada bidang *software* dan *hardware*.tulisan yang pernah dibuat adalah Seni Kriptografi dan Algoritma Enkripsi data Sederhana dengan menggunakan Visual Basic 6.0(paradigma vol X No 1.januari 2008,ISSN 1410-5963.pada saat ini penulis sedang melanjutkan pendidikan pasca sarjana ilmu komputer STMIK Nusa Mandiri Jakarta.

Rachmat Adi Purnama, Tempat dan tanggal lahir di Jakarta , 26 Nopember 1970 dengan riwayat pendidikan S1 STMIK Budi Luhur lulus 1997, pernah membuat penulisan jurnal MERAMAL MASA DEPAN MIKROPROSESOR. Jurnal Pardigma Vol. II No. 2 Desember 1999 dan PERAN PENTING PROTOCOL DALAM KOMUNIKASI DATA PADA SISTEM JARINGAN KOMPUTER. Jurnal Paradigma Vol. IV No. 1 Januari 2001 serta pernah membuat buku ajar Manajemen Proyek penerbit BSI 1999 dan Komunikasi Data penerbit BSI 1998. Sedangkan kegiatan yang pernah diikuti antara lain: Aktif mengikuti kegiatan-kegiatan seminar baik sebagai peserta maupun pembicara dan pekerjaan yang ditekuni saat ini adalah sebagai Staf Akademik pada Bina Sarana Informatika terhitung Agustus 1997 sampai dengan sekarang.

Mochamad Wahyudi, Lulusan Magister Ilmu Komputer (M.Kom) 2008 Konsentrasi Rekayasa *e-Bisnis* pada Universitas Budi Luhur, Jakarta dan lulusan Magister Manajemen (MM) 2003 Konsentrasi Manajemen Sistem Informasi Universitas Budi Luhur, Jakarta. Beliau banyak membuat penulisan ilmiah yang diterbitkan di jurna-jurnal, antara lain : Teknologi Telepon Menggunakan *Internet Protocol (Voice Over Internet Protocol).* Jurnal Ilmiah Paradigma, Bina Sarana Informatika. ISSN : 1410 – 5963.Vol. V No. 3. September 2002. Pengaturan *Trafic Light* Berbasis *Fuzzy Logic* menggunakan Komputer IBM PC - XT 8088. Jurnal Ilmiah Padma. Universitas Budi Luhur. ISSN : 1410 – 819. 04 April 2002, Perancangan sistem Penunjuk (Indikator) Posisi Pita Kaset. Jurnal

Ilmiah Paradigma. Bina Sarana Informatika . ISBN : 1410 – 5963. Vol. II No. 2. April 2001 dan Perancangan Sistem Pengembangan Film Menggunakan IBM PC - XT 808. Jurnail Ilmiah Paradigma. Bina Sarana Informatika. ISSN : 1410 – 5963. Vol. IV No. I. Januari 2001