

# Keamanan Pada Perancangan Basisdata

Oleh :

**Janner Simarmata**

**[sijanner@yahoo.com](mailto:sijanner@yahoo.com)**

**<http://simarmata.cogia.net>**

[Dept of Computer Science - University of Gadjah Mada -Yogyakarta]

**Iman Paryudi**

**[paryudi@yahoo.com](mailto:paryudi@yahoo.com)**

[Dept of Computer Science - Nowrosjee Wadia College, University of Pune – Mumbai, India]

14 Januari 2006

*Dipublikasikan dan didedikasikan  
untuk perkembangan pendidikan di Indonesia melalui*

**MateriKuliah.Com**

***Lisensi Pemakaian Artikel:***

*Seluruh artikel di **MateriKuliah.Com** dapat digunakan, dimodifikasi dan disebarakan secara bebas untuk tujuan bukan komersial (nonprofit), dengan syarat tidak menghapus atau merubah atribut Penulis. Hak Atas Kekayaan Intelektual setiap artikel di **MateriKuliah.Com** adalah milik Penulis masing-masing, dan mereka bersedia membagikan karya mereka semata-mata untuk perkembangan pendidikan di Indonesia. **MateriKuliah.Com** sangat berterima kasih untuk setiap artikel yang sudah Penulis kirimkan.*

## 1. Pendahuluan

Supaya basisdata yang kita rancang mendukung perusahaan dalam mencapai tujuannya, basisdata harus bisa diakses dan di update oleh pengguna. DBA dan pembuat aplikasi harus memberi pengguna akses ke basisdata yang mereka butuhkan untuk menyelesaikan pekerjaan mereka, sembari meminimalkan kerusakan yang dapat mereka lakukan pada sistem dan menyembunyikan data yang tidak boleh dilihat oleh mereka.

## 2. Pentingnya Keamanan pada Perancangan Basisdata

Untuk pengguna yang mengakses sistem, harus bisa mengakses melalui *Local Area Network* (LAN) atau *Wide Area Network* (WAN) dan sekarang ini yang paling banyak melalui *World Wide Web*. Aplikasi menggunakan aplikasi web browser sebagai antarmuka utama sudah umum sekarang ini. Ketika kita meletakkan basisdata pada web, dia menjadi mudah diserang oleh hacker dan penjahat lainnya dari luar organisasi yang akan merusak atau mencuri data. Bayangkan bahwa gaji semua orang, data personalia, atau data rahasia perusahaan lainnya ditampilkan kepada umum melalui website. Bahkan dari dalam perusahaan sendiri dapat terjadi seorang pegawai merusak sistem ketika mereka dipecat.

Kebanyakan basisdata sekarang ini memungkinkan akses melalui WWW. Bagian dari perancangan harus memasukkan obyek-obyek basisdata (pengguna, kode, tabel, role) yang mendukung akses web dan syarat keamanan yang vital seperti sertifikat dan SSL yang harus disertakan dengan akses tersebut.

Kemungkinan terjadinya gangguan dari orang-orang baik dalam maupun luar organisasi harus dilawan. Masalah yang lebih besar dapat terjadi ketika orang-orang dalam perusahaan harus menanggulangi kemungkinan kesalahan yang tak disengaja. Menghapus atau menimpa suatu file data, menghapus tabel atau mengupdate kolom secara tidak benar dengan tidak sengaja, dapat menimbulkan masalah seperti yang ditimbulkan oleh hacker. Oleh karena itu keamanan harus direncanakan dengan baik dan diintegrasikan dalam basisdata.

Basisdata seharusnya tidak hanya menyediakan data pada pengguna tapi juga menyediakan proteksi pada data tersebut.

### **3 Yang Berhak Mengakses Basisdata**

Sebagian informasi yang dibutuhkan selama perancangan dan penerapan akan dikumpulkan dari wawancara selama analisis. Supaya dapat merencanakan keamanan basisdata selama perancangan basisdata dan melaksanakan keamanan setelah penerapan, semua pengguna basisdata harus ditetapkan terlebih dahulu. Ada beberapa kategori pengguna untuk setiap sistem informasi, mulai dari pengguna akhir sampai administrator sistem informasi. Pengguna yang mempunyai akses untuk sistem informasi adalah sebagai berikut:

- Pengguna akhir
- Pelanggan
- Manajemen
- Administrator jaringan
- Administrator sistem
- Administrator basisdata
- Pemilik skema

Masing-masing pengguna ini mempunyai tingkatan akses yang berbeda. Cara termudah untuk menerapkan keamanan adalah dengan memberi semua pengguna akses maksimum, tapi kelemahannya adalah adanya kemungkinan kehilangan atau penyalagunaan data. Pengguna basisdata harus mempunyai akses yang cukup untuk melaksanakan pekerjaannya. Dengan kata lain akses yang diberikan kepada pengguna harus sekecil mungkin untuk menghindari masalah.

Beberapa informasi bisa merupakan informasi rahasia atau sensitif. Jika ada data yang sensitif, sebaiknya data diisolasi dengan memecahnya menjadi tabel terpisah yang dapat dihubungkan dengan tabel-tabel yang tidak sensitif atau dengan membuat view yang tidak memasukkan data yang sensitif.

Pengguna yang ingin melihat semua data akan diberi akses untuk suatu tabel, dan pengguna yang tidak mempunyai akses hanya diberi hak untuk melihat view saja.

Ketika hak akses pengguna pada suatu data ditentukan, tipe akses harus ditentukan. Secara umum data dapat dibuat, diambil, di update, dan dihapus dari basisdata. Beberapa pengguna bisa jadi ingin melakukan semua operasi pada basisdata. Pengguna lainnya mungkin hanya ingin melakukan query pada basisdata, atau mengupdate data dari tabel tertentu dalam basisdata. Pengguna tertentu tidak diperbolehkan untuk menghapus data.

### **3.1 Tingkatan Akses**

Pengguna basisdata diberi identitas pengguna dengan password dan kemampuan untuk berhubungan dengan basisdata. Basisdata modern membutuhkan keputusan tentang perancangan yang berhubungan dengan identitas pengguna dan password. Keamanan digunakan untuk mengontrol akses. Pengguna mempunyai akses ke baik data maupun sumberdaya, keduanya dapat dikontrol pada tingkatan basisdata.

Apabila pengguna diberi identitas pengguna, mereka dapat mengakses basisdata secara langsung kedalam jaringan melalui aplikasi client-server. Mereka juga dapat log in melalui aplikasi middle tier untuk mengakses basisdata. Sebagai contoh, pengguna dapat log in ke server aplikasi web yang mempunyai banyak pengguna web yang log in ke server tersebut. Server aplikasi web kemudian berkomunikasi dengan basisdata dengan cara log in dengan menggunakan identitas pengguna yang mempunyai akses minimum yang dibutuhkan untuk menyelesaikan pekerjaan yang sedang dikerjakan oleh aplikasi web. Aplikasi client-server umumnya hanya mempunyai satu identitas pengguna untuk masing-masing pengguna. Aplikasi web yang menggunakan server aplikasi web middle tier dapat menggunakan satu atau beberapa pengguna basisdata. Pada kedua kasus tersebut, bagian dari perancangan

basisdata adalah menentukan kombinasi privilege Create, Retrieve, Update, Delete (CRUD-atau INSERT, SELECT, UPDATE, DELETE dalam SQL) pada kombinasi apa dari tabel basisdata yang dibutuhkan untuk menyelesaikan masing-masing pekerjaan yang didukung oleh basisdata. Masing-masing kombinasi privilege yang dibutuhkan untuk menyelesaikan pekerjaan tertentu harus ditentukan dan kemudian dianalisa untuk memastikan bahwa akses yang diberikan tidak melebihi yang dibutuhkan untuk menyelesaikan pekerjaan.

Pada beberapa basisdata, pengguna dapat log in ke basisdata berdasarkan otorisasi eksternal. Pada otorisasi eksternal pengguna mempunyai ID pengguna pada sistem operasi yang dibuat pada Unix, Novell atau NT untuk basisdata. tersebut. Pengguna yang mempunyai identitas sistem operasi merupakan pengguna basisdata yang diatur untuk otorisasi eksternal. Menggunakan otorisasi eksternal, pengguna sistem operasi yang log in ke basisdata dapat secara otomatis log in ke basisdata tanpa password. Pengguna sistem operasi dan basisdata yang diatur untuk otorisasi eksternal biasanya adalah DBA atau account basisdata yang menjalankan program eksternal seperti program C++ yang mengakses basisdata.

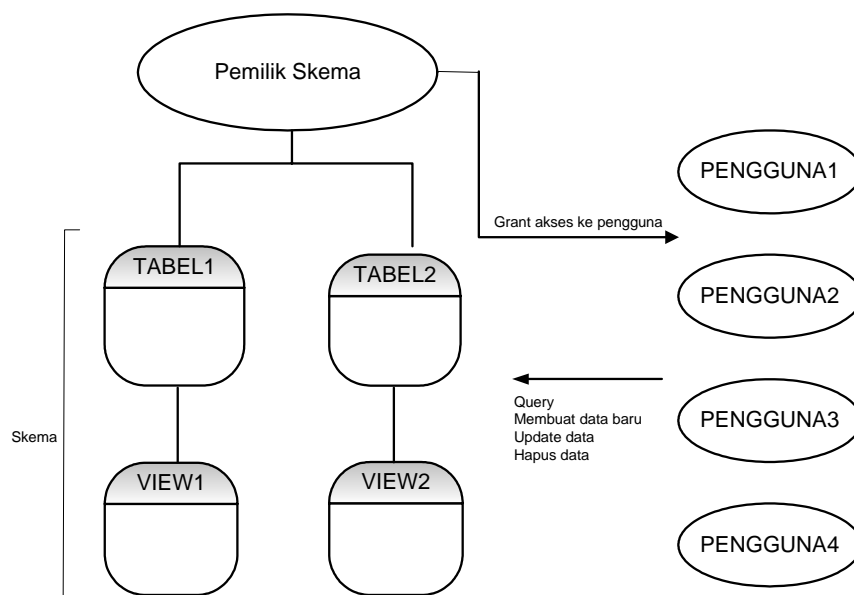
Sistem Administrator sistem operasi (SA) (root pada Unix, Admin pada Novell) dapat mengontrol account sistem operasi ini dan juga account basisdata. Baik *Sistem Administrator* dan *Administrator Basisdata* mempunyai potensi melakukan kesalahan. Bagian dari proses adalah *cek and balance* untuk memastikan bahwa account yang super ini tidak disalahgunakan.

Berikut ini adalah daftar yang menunjukkan tingkatan akses ke suatu sistem informasi :

- Pengguna Super (root pada Unix, Admin pada Novell)
- Pemilik Basisdata
- Pemilik Skema
- Pengguna Akhir

**Pengguna super** adalah account pada sistem operasi yang mempunyai privilege yang paling banyak. Pengguna super mempunyai akses ke semua file yang disimpan pada sistem. **Pemilik basisdata** mempunyai akses ke semua file yang berhubungan dengan software basisdata dan file data pada suatu sistem. Meskipun pemilik basisdata dibatasi pada file-file yang berhubungan dengan basisdata, tapi perlu diingat bahwa sebagian file pada sebagian sistem adalah berhubungan dengan basisdata. **Pemilik skema** adalah pembuat dan pemilik obyek-obyek basisdata yang digunakan untuk aplikasi pengguna. Pemilik skema mempunyai akses tak terbatas ke seluruh obyek skema dan bertanggungjawab untuk mengontrol akses tersebut ke account pengguna lainnya. **Pengguna akhir** mempunyai akses yang paling sedikit, meskipun basisdata dibuat untuk pengguna akhir.

**Gambar 1.1** menunjukkan hubungan pengguna dengan basisdata. Obyek-obyek basisdata dipunyai oleh pemilik skema yang memberi akses terhadap obyek tersebut kepada pengguna akhir. Jika pengguna akhir mempunyai privilege yang cukup, mereka dapat mengakses tabel, view, dan obyek lainnya didalam skema.



**Gambar 1.1** Hubungan pengguna dengan basisdata

Pengguna dalam suatu basisdata dapat juga mengakses basisdata lainnya dengan cara menghubungkan dua basisdata tersebut. Hubungan ini akan menghubungkan pengguna ke basisdata pertama ke pengguna pada basisdata kedua, dan pengguna pada pertama kemudian dapat mengakses basisdata kedua dengan privilege yang dipunyai oleh pengguna pada basisdata kedua. Hubungan basisdata ini merupakan tool yang sangat berguna, tapi harus diatur untuk memastikan mereka tidak disalahgunakan. Dapat dengan mudah untuk melihat seseorang pengguna suatu basisdata dengan menghubungkannya ke basisdata lainnya. Hubungan ini dapat dibuat oleh account DBA.

### 3.2 Privilege

**Privilege** digunakan untuk mengontrol akses pengguna. Privilege terdapat pada tingkatan sistem operasi, basisdata, dan aplikasi. Privilege basisdata mengontrol akses pengguna dalam lingkungan basisdata, seperti manipulasi struktur basisdata dan akses ke obyek skema. Ada dua tipe dasar privilege dalam basisdata relasional:

- Privilege sistem
- Privilege obyek

**Privilege sistem** terdiri dari hal-hal yang memungkinkan pengguna untuk melakukan tugasnya pada ruang lingkup basisdata, sedangkan **privilege obyek** memungkinkan pengguna untuk melakukan tugasnya pada ruang lingkup skema. Privilege sistem biasanya termasuk kemampuan untuk membuat tabel, menghapus tabel, merubah struktur tabel, membuat indeks dan view, dan memanipulasi account pengguna. Privilege obyek termasuk kemampuan untuk mengambil data dari tabel dan memanipulasi data tabel. Privilege sistem berbeda-beda antara satu perangkat lunak basisdata relasional dengan yang lainnya. Sedangkan privilege obyek lebih standar.

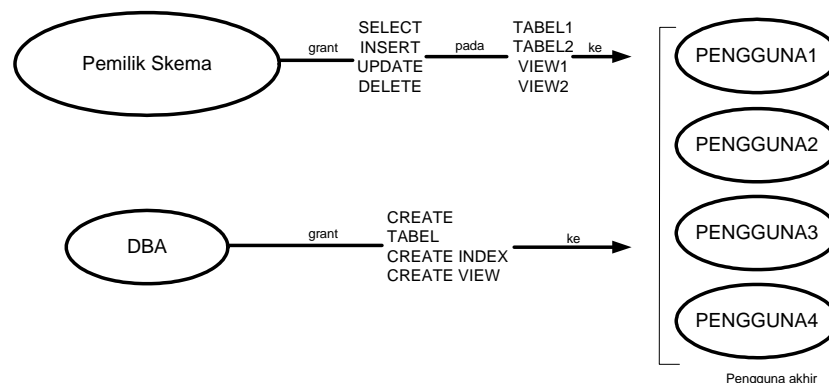
Berikut adalah privilege obyek untuk basisdata relasional

- **SELECT** – memungkinkan data diambil dari tabel

- **INSERT** - memungkinkan pembentukan baris data baru pada tabel
- **UPDATE** - memungkinkan data yang sudah ada dalam tabel untuk dimodifikasi
- **REFERENCES** - memungkinkan kolom dalam tabel untuk diacu oleh kolom lain (*seperti melalui kunci tamu*)
- **USAGE** - memungkinkan penggunaan domain tertentu

Privilege di grant kepada pengguna dengan perintah GRANT dan dicabut dari pengguna dengan perintah REVOKE. Dua pilihan dapat digunakan dengan perintah GRANT untuk memungkinkan pengguna lain memberi grant pada pengguna lainnya. Jika seorang pengguna di grant privilege obyek dengan WITH GRANT OPTION, dia diberi hak akses khusus ke obyek basisdata, tapi juga dapat meng-grant akses tertentu terhadap obyek basisdata kepada pengguna lain, meskipun pengguna asli tidak mempunyai obyek. Demikian juga dengan WITH ADMIN OPTION yang dapat ditambahkan pada perintah GRANT yang berhubungan dengan privilege sistem, dan memungkinkan seorang pengguna untuk meng-grant privilege sistem ke pengguna lain.

**Gambar 1.2** mengilustrasikan proses pengaturan akses pengguna akhir melalui privilege basisdata. Dua tipe privilege ditunjukkan pada gambar yaitu privilege obyek dan privilege sistem. Pemilik skema bertanggungjawab untuk meng-grant privilege obyek kepada pengguna, dan DBA meng-grant privilege sistem.



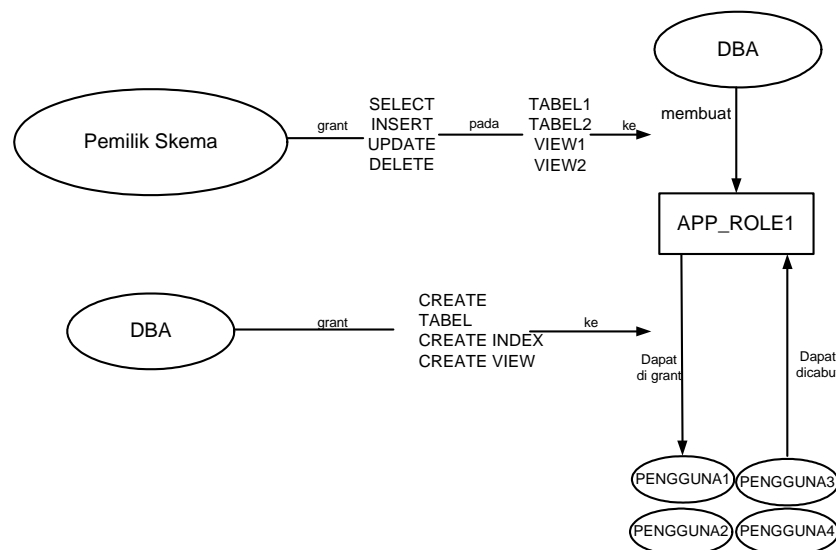
**Gambar 1.2** Proses pengaturan akses pengguna akhir melalui privilege basisdata



### 3.3 Roles

Kombinasi dari privilege basisdata dan privilege tabel CRUD yang diperlukan untuk menyelesaikan pekerjaan tertentu dapat diberikan pada pengguna secara individu. Tetapi hal ini tidak efektif untuk dilakukan. Sebagai bagian dari perancangan, kita bisa menentukan fungsi pekerjaan apa yang ada pada aplikasi kita dan privilege apa yang ada pada basisdata, tabel-tabelnya, dan obyek-obyek basisdata lain yang dibutuhkan oleh masing-masing fungsi pekerjaan. Kita kemudian bisa membuat obyek basisdata yang disebut role, yang mengikutsertakan seluruh privilege yang dibutuhkan untuk menyelesaikan fungsi tersebut.

Pengguna kemudian di grant satu atau lebih sejumlah role. Identitas pengguna di grant pada role tersebut yang dibutuhkan untuk menyelesaikan semua fungsi yang berhak dilakukan oleh pengguna tertentu. Sebagai tambahan pada SELECT, INSERT, UPDATE, atau DELETE, role bisa juga memasukkan privilege untuk merubah obyek, membuat indeks pada obyek, menghubungkannya melalui kunci tamu, atau mengeksekusinya. Menggunakan role menyederhanakan pengaturan akses basisdata dan pengaturan privilege obyek basisdata.



**Gambar 1.3** Pemakaian Role Basisdata

**Gambar 1.3** mengilustrasikan pemakaian role basisdata. Pertama DBA membuat role. Ketika role dibuat, dia merupakan obyek kosong yang hanya mempunyai nama. Role tersebut dapat di grant ke pengguna setiap saat setelah dia dibuat, meskipun privilege biasanya di grant ke role terlebih dahulu. Privilege obyek dapat di grant oleh role ke pemilik skema dan privilege sistem dapat di grant oleh DBA. Role dapat disebut sebagai kumpulan privilege yang dapat di grant dan dicabut dengan mudah pada pengguna basisdata. Seorang pengguna dapat di grant satu atau banyak role, tergantung pada pembagian privilege untuk suatu aplikasi dan tingkatan akses dari pengguna. Ketika log in ke basisdata, pengguna mempunyai semua privilege yang terkandung pada role yang telah di grant pada role tersebut.

#### **4. Yang Bertanggungjawab Terhadap Keamanan**

Tidak ada satu orang yang bertanggungjawab penuh terhadap keamanan. Semua pengguna harus bertanggungjawab pada penerapan, pelaksanaan dan kepatuhan terhadap keamanan. Sistem administrator, DBA dan pemilik skema bertanggungjawab pada penerapan dan pelaksanaan. Semua pengguna harus patuh pada keamanan. Pengguna harus dapat dipercaya untuk menggunakan data yang mereka akses dengan benar. Administrator harus membagi akses ke data secara hati-hati dan aturan harus diikuti oleh semua pengguna.

Aturan keamanan dibuat oleh manajemen dan staf teknis. Aturan ini harus digunakan sebagai acuan selama perencanaan dan penerapan sistem informasi. Staf teknis yang turut serta dalam pembuatan aturan keamanan termasuk administrator jaringan, administrator sistem, dan administrator basisdata. Ada tiga tingkatan dasar pengaturan keamanan yang berhubungan dengan sistem informasi:

- Pengaturan tingkat sistem.
- Pengaturan tingkat basisdata.
- Pengaturan tingkat aplikasi.

Tingkatan akses pengguna dikontrol oleh administrator sistem operasi, administrator basisdata, dan administrator aplikasi. Biasanya pengguna akhir harus log in ke sistem operasi untuk mengakses aplikasi, yang selanjutnya mengakses basisdata. Baik aplikasi maupun basisdata terletak pada lingkungan sistem operasi. Tapi sekarang sebagian besar pengguna melewati otentifikasi sistem operasi dan langsung menuju basisdata dari aplikasi yang diinstall pada PC atau melalui web browser pada PC.

Tergantung pada kebutuhan pengguna akhir, tingkatan akses yang berbeda dapat di grant . Sebagian besar pengguna harus berurusan dengan kombinasi otentifikasi basisdata dan aplikasi. Misalkan seorang pengguna mengakses basisdata melalui aplikasi Visual Basic yang diinstall pada PC. Pengguna tersebut dapat diberi akses ke basisdata yang di grant oleh pemilik skema. Aplikasi tergantung pada privilege yang telah diberikan pada pengguna dalam basisdata. Jika otentifikasi pengguna ditangani pada tingkatan aplikasi, keamanan pada aplikasi tersebut lemah dan merupakan perancangan yang buruk. Oleh karena itu otentifikasi pengguna harus ditangani pada tingkatan basisdata, menggunakan privilege dan role. Keamanan pada aplikasi adalah tambahan dan hanya digunakan untuk meningkatkan keamanan basisdata.

#### **4.1 Pengaturan tingkat sistem.**

Administrator sistem yang disebutkan dimuka mengontrol host komputer untuk basisdata dan file-file yang terdapat dalamnya. Ada kemungkinan mereka merusak file-file data baik tidak disengaja maupun disengaja.

Account yang dimiliki oleh sistem operasi akan dibuat untuk file-file basisdata. Orang yang mempunyai account semacam itu dapat melakukan kesalahan yang sama seperti yang dilakukan oleh administrator sistem dengan manghapus atau menimpa file basisdata atau menyalin data secara tidak tepat.

Pada lingkungan Unix, privilege file basisdata pada sistem operasi adalah masalah keamanan utama dan harus dimonitor secara ketat. Pada unix, account

pengguna sistem operasi yang merupakan pemilik basisdata harus dapat membaca, menulis, mengeksekusi semua file basisdata. Pada lingkungan Windows NT, account administrator akan mempunyai **“Full Control”** sementara pengguna local membutuhkan akses **“Read”** untuk mengeksekusi program dalam basisdata seperti SQL.

Pengguna selain pemilik basisdata seharusnya hanya mempunyai privilege terbatas untuk menghindari kemungkinan perubahan atau penghapusan file. Pastikan bahwa privilege untuk akses file pada tingkat sistem operasi diset secara benar, tidak ada privilege lain yang diberikan selain yang dibutuhkan.

Keamanan fisik dari basisdata adalah masalah vital. Ruang komputer dengan akses terbatas dan kunci yang baik sangat dibutuhkan. Tergantung pada tingkat kepentingan data, penjaga dan sistem keamanan dapat juga disertakan. Hal yang sama juga bias dilakukan pada server web atau server aplikasi dan komponen jaringan utama seperti hub dan router. Sistem backup dengan tape yang disimpan ditempat lain juga diperlukan dalam hal keamanan fisik tidak berjalan dengan baik.

Menyalin file data dari basisdata baik ke drive lain atau ke tape dapat dilakukan ke sistem lainnya dan diinstal. Basisdata yang disalin ke host yang baru ini dapat diakses oleh administrator sistem dan DBA dari host yang baru tersebut. Harus ada prosedur untuk mengontrol tape yang digunakan untuk backup. Jika datanya rahasia, tape tersebut harus diamankan.

#### **4.2 Pengaturan tingkat Basisdata**

DBA mengontrol dan mengatur basisdata. Pada analisis tahap akhir, DBA dapat melakukan apapun yang dia ingin lakukan. DBA juga bertanggungjawab untuk memastikan bahwa basisdata mempunyai memori dan ruang disk yang cukup dan dikonfigurasi secara benar. DBA mengontrol keamanan basisdata, meskipun DBA dapat melimpahkan tugas pengamanan pada orang lain, sedang pemilik

tabel dan obyek basisdata lainnya mempunyai kontrol langsung pada siapa yang dapat mengakses obyek-obyek tersebut.

DBA mirip dengan polisi. Dia mempunyai kekuasaan yang besar dan melakukan banyak hal termasuk menyediakan infrastruktur dan lingkungan keamanan, tapi tidak berusaha mengontrol apa yang terjadi pada lingkungan itu kecuali kalau ada masalah. Pekerjaannya yang pertama adalah mencegah masalah dengan menyediakan sumberdaya dan melaksanakan prosedur keamanan yang baik, dan yang kedua adalah menyelesaikan masalah secara cepat ketika timbul masalah.

Penting untuk membuat keamanan pada tingkat basisdata, tidak hanya keamanan pada tingkat aplikasi. Membuat role pengguna dalam basisdata dan kemudian membagi kemampuan mereka melalui keamanan aplikasi hanya berkerja untuk aplikasi tersebut, tetapi banyak aplikasi mengakses obyek-obyek basisdata itu.

#### **4.3 Pengaturan Tingkat Aplikasi**

Pembuat aplikasi menentukan struktur tabel, menulis program, dan menentukan role yang memungkinkan pengguna untuk melakukan CRUD pada tabel dan mengeksekusi program.

Aplikasi dapat mengikutsertakan keamanannya sendiri sebagai suatu tambahan pada keamanan basisdata. Tabel dapat dibuat pada aplikasi yang memiliki skema untuk menyimpan informasi pengguna tertentu dan role aplikasi dapat ditentukan dan kemudian dikonfigurasi berdasarkan tabel-tabel tersebut. Pada dasarnya aplikasi membatasi pilihan untuk pengguna tertentu berdasarkan nilai-nilai pada tabel aplikasi atau nilai dari beberapa kelompok variabel dalam aplikasi.

Pembuat aplikasi memutuskan privilege apa dan pada tabel apa harus dimasukkan pada masing-masing role atau kelompok privilege yang ditempatkan pada basisdata. Jika pengguna tertentu tidak mempunyai privilege

untuk mengakses tabel atau mengeksekusi program, dia tidak dapat melakukannya. Penentuan role yang benar adalah salah satu aspek penting dalam keamanan. Terlalu sedikit privilege dalam suatu role akan menyebabkan pengguna tidak dapat melakukan pekerjaannya. Terlalu banyak privilege maka pengguna dapat melihat informasi terlalu banyak dan dapat merusak data.

Perangkat lunak aplikasi dan basisdata modern sekarang memasukkan fasilitas untuk otentifikasi biometric dari pengguna pada saat log in melalui sidik jari, scan mata atau suara. Kartu keamanan atau kunci fisik lainnya juga dimasukkan dalam kemampuan otentifikasi log in basisdata.

## **5. Menggunakan View dan Prosedur untuk Meningkatkan Keamanan**

View dan stored procedure adalah mekanisme yang bergubungan dengan basisdata relasional yang membantu meningkatkan fleksibilitas penerapan keamanan. Obyek-obyek ini sangat berguna untuk membatasi akses ke data dan menyaring bagian data yang dapat diakses oleh pengguna.

Penggunaan view sangat umum pada tool keamanan pada tingkat aplikasi. Sebuah view adalah tabel logis. Maksudnya adalah view merupakan query pada tabel fisik yang telah diberi nama dan disimpan pada basisdata sebagai obyek basisdata terpisah. Privilege kemudian dapat di grant pada pengguna untuk view tersebut bukan pada tabel. View kemudian dapat dipilih oleh pengguna dengan privilege tertentu seolah-olah view tersebut adalah tabel.

View memungkinkan pembuat aplikasi untuk membuat tabel logis dengan hanya menampilkan kolom tertentu yang dapat dilihat oleh pengguna untuk melaksanakan pekerjaannya pada saat tertentu. Ini memungkinkan pembuat aplikasi untuk menyembunyikan kolom-kolom tabel yang tidak boleh dilihat oleh pengguna. Pengguna hanya mempunyai akses ke view, bukan ke tabelnya. Yang bisa dilihat pengguna hanyalah kolom pada view, sedang kolom pada tabel tidak dapat dilihat oleh pengguna.

Pengguna hanya dapat melihat data dalam format yang diinginkan oleh pembuat dengan nama kolom sesuai dengan pilihan pembuat. Pada kondisi tertentu, data pada tabel dibelakang view dapat di update melalui view.

Pembuatan *stored procedure* juga merupakan metoda yang umum digunakan untuk meningkatkan keamanan secara keseluruhan. Prosedur dapat dibuat untuk melakukan operasi terhadap satu atau lebih tabel dalam basisdata. Pengguna yang mengeksekusi prosedur hanya dapat melakukan aksi seperti program yang tertulis didalamnya. Sebagai contoh prosedur dibuat untuk mengatur data karyawan. Prosedur memungkinkan pegawai baru dimasukkan, demikian juga perubahan data pegawai seperti alamat. Prosedur tidak memungkinkan data pegawai dihapus dari basisdata. Seperti pada view, akses di grant ke prosedur dan bukan pada tabel.

Pengguna hanya dapat melakukan aktifitas yang diperbolehkan oleh prosedur, yang dieksekusi dibawah privilege pemiliknya. Prosedur semacam ini bisa menghalangi pengguna untuk menghapus semua record dalam tabel.

## **6. Perancangan Sistem Manajemen Keamanan.**

Meskipun ada fitur keamanan built in seperti *role*, beberapa pengembang tetap membuat sistem pengaturan keamanan. Sistem pengaturan keamanan biasanya meliputi perancangan dan pembuatan tabel basisdata yang digunakan oleh aplikasi untuk menentukan akses pengguna tertentu ke data. Sistem pengaturan keamanan diperlukan hanya jika fitur built in dari perangkat lunak basisdata yang digunakan tidak memenuhi kebutuhan keamanan organisasi. Sistem keamanan seperti ini biasanya hanya sebagai tambahan fitur yang ada pada perangkat lunak basisdata.

## **7. Tindakan Pencegahan Tambahan**

Saat ini basisdata dan jaringan berhubungan erat. Dengan perkembangan aplikasi berbasis web, trend ini semakin berkembang. Keamanan pada web

sangat utama pada sebagian besar aplikasi. Komunikasi internet basisdata hampir selalu berdasarkan pada protokol TCP/IP. TCP/IP adalah protokol komunikasi data standar untuk web. Sayangnya TCP/IP sendiri tidak mempunyai pengamanan. Data anda dan nomor kartu kredit seseorang dapat dicuri apabila menggunakan TCP/IP. Pengamanan harus ditambahkan pada TCP/IP dari sumber lain untuk membuat basisdata web modern lebih aman.

### 7.1 Keamanan Jaringan

Keamanan jaringan adalah barisan pertahanan paling depan. Jika anda dapat menghindarkan orang jahat mengambil alamat IP basisdata anda, anda telah dapat menjauhkan mereka dari basisdata anda. Ini disebut dengan menyembunyikan. *Network Address Translation (NAT)* akan menterjemahkan alamat IP asli yang digunakan sistem anda dan menyembunyikannya dari pengguna web. NAT dimasukkan pada sebagian besar produk firewall dan akan membuat hacker tidak berpengalaman tidak bisa memasuki sistem anda. Sayangnya menyembunyikan tidak menghentikan hacker berpengalaman, ini hanya memperpanjang proses.

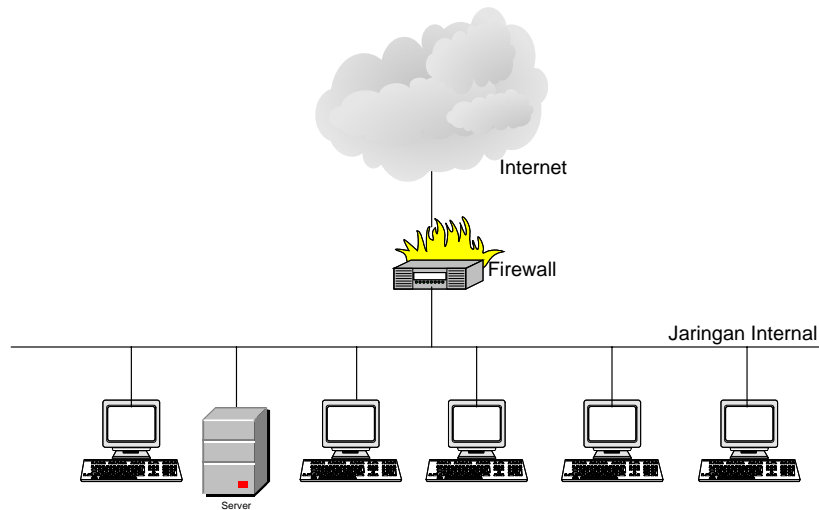
Memiliki file konfigurasi jaringan workstation yang berbeda tergantung pada kebutuhan kelompok tertentu adalah ide yang bagus. Setidaknya anda tidak menempatkan alamat-alamat IP untuk seluruh perusahaan dalam satu file.

### 7.2 Firewall Jaringan

**Firewall jaringan** adalah pertahanan anda yang sebenarnya dari hacker. Setting firewall yang benar akan melindungi jaringan anda, tapi ini juga dapat memperlambat kinerja jaringan dimana aplikasi anda berjalan sangat lambat. Konfigurasi firewall yang tidak tepat akan menyebabkan operasi berbasis web anda berhenti. Oleh karena itu diperlukan orang yang mengetahui apa yang harus dilakukan untuk merawat firewall jaringan.[Brent]



Sebuah firewall memberikan cara pada suatu organisasi untuk membuat ruang antara jaringan yang terisolasi dari jaringan luar, seperti internet, dan jaringan yang terhubung dengan jaringan luar. Firewall menyediakan cara mudah untuk mengontrol jumlah dan macam lalu lintas yang akan melewati ruang antara jaringan internal dalam organisasi dan jaringan luar.



**Gambar 1.4** Sebuah firewall pada umumnya memisahkan jaringan internal dari Internet

Firewall jaringan adalah kombinasi perangkat lunak dan perangkat keras yang dimaksudkan untuk mengusir hacker dari jaringan anda tapi mengijinkan orang-orang yang mempunyai akses. Firewall menyaring paket data TCP/IP masuk dan keluar berdasarkan pada sumber dan tujuan alamat IP dan pada fungsi yang dikandung dalam pembawa paket data. Router jaringan mempunyai daftar alamat IP sumber dan tujuan yang diperkenankan. Paket dengan alamat yang tidak ada dalam daftar tidak diperbolehkan masuk. Jaringan utama anda, dimana basisdata anda biasanya terletak, akan berada dibelakang firewall dengan daftar alamat yang diperkenankan yang sangat terbatas.

### 7.3 Secure Socket Layer (SSL)

Kemampuan tambahan yang dapat digunakan untuk mengamankan TCP/IP adalah *Secure Socket Layer (SSL)* atau *Internet Engineering Task Force Transport Layer Security (IETF TLS)*.

SSL menyediakan enkripsi data sehingga data tidak dapat dirusak ketika data itu bergerak melalui router internet. Sistem enkripsi sering menggunakan sistem kunci rahasia yang digunakan untuk mengacak data pada waktu transmisi pengiriman dan kemudian mengembalikan data ke keadaan semula sesampainya ditujuan.

SSL menggunakan metoda enkripsi kunci publik. Sistem SSL menghasilkan pasangan kunci, satu untuk publik satu untuk pribadi, yang secara bersama-sama dapat digunakan untuk mengenkripsi atau uncrypt data. Kunci publik digunakan untuk berkomunikasi melalui web dan kunci publik itu digunakan untuk mengenkripsi data yang dikirimkan orang lain dan *uncrypt* data yang kita kirimkan. **Kunci pribadi** digunakan untuk mengenkripsi data yang anda kirimkan dan untuk meng-uncrypt data yang dikirimkan pada anda yang dienkripsi oleh orang lain dengan menggunakan publik kunci. Enkripsi dengan kunci publik relatif lambat, sehingga SSL menggunakannya selama pembentukan sesi awal dan kemudian menghasilkan dan bertukar kunci rahasia yang disebut kunci sesi yang digunakan oleh kedua pihak untuk sesi selanjutnya. Enkripsi kunci rahasia relatif cepat dibandingkan enkripsi kunci publik.

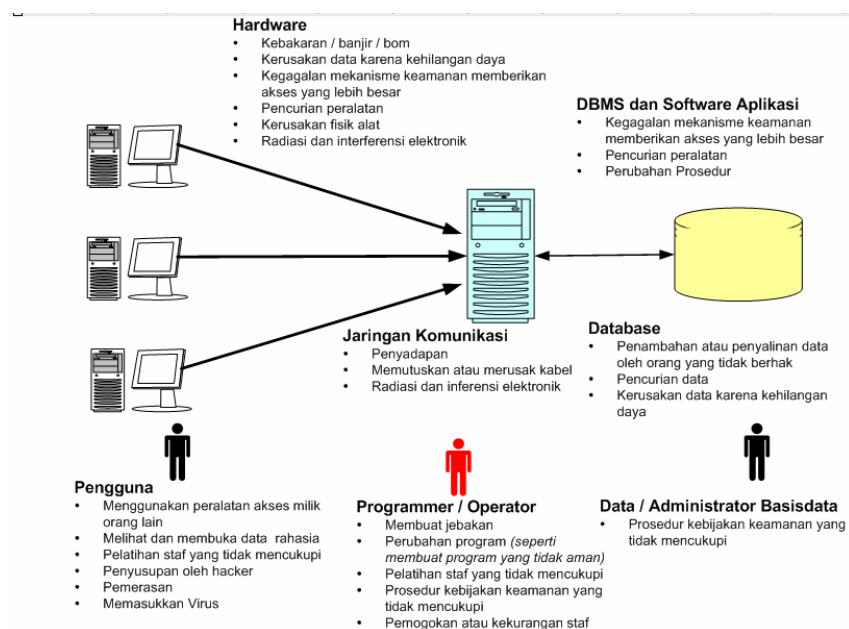
Bagian dari setting website yang aman adalah mendapatkan surat kepercayaan situs elektronik yang disebut sertifikat dari perusahaan pihak ketiga yang terpercaya yang disebut pihak yang berwenang untuk memberi sertifikat (**Certifying Authority**). Certifying Authority akan memvalidasi bahwa anda termasuk organisasi yang sah, sebelum mereka memberikan sertifikat pada anda. Sesi komunikasi pengguna anda kemudian mengakses sertifikat anda sebagai langkah otentifikasi awal ketika mereka mengakses basisdata anda

melalui web. SSL juga memvalidasi integritas data pada kedua ujung komunikasi.

## 8 Pelanggaran Pada Keamanan

Pelanggaran pada keamanan biasanya diketahui melalui pengamatan basisdata secara teratur atau selama pemecahan masalah. Pelanggaran bisa disengaja maupun tidak disengaja.

- Pelanggaran disengaja biasanya ditimbulkan oleh maksud jahat atau keinginan untuk melihat-lihat.
- Pelanggaran tak disengaja dapat terjadi secara tidak sengaja memasuki layar dalam aplikasi dimana pengguna itu tidak diperbolehkan untuk mengaksesnya (*kadang-kadang administrator membuat kesalahan ketika menerapkan privilege*). Pelanggaran dapat berasal dari luar organisasi atau dari dalam perusahaan. Misalkan seorang pengguna mempunyai akses ke data tertentu dan secara sengaja membuat perubahan terhadap data. Ini adalah sebuah contoh pelanggaran. Contoh lainnya adalah usaha untuk menghack kedalam sebuah basisdata.



**Gambar 1.5** Penyerangan terhadap Sistem Komputer

Berikut adalah langkah-langkah proaktif yang dapat dilakukan untuk mencegah pelanggaran keamanan:

- Merubah password administrator secara teratur.
- Memaksa pengguna untuk sering merubah password.
- Melarang penggunaan password secara bersama.
- Menghapus account pengguna yang tidak aktif.
- Menghapus account pengguna yang bukan pegawai.
- Melakukan pengawasan acak terhadap semua aktifitas.
- Melakukan audit basisdata.
- Memberi pengertian pada pengguna akhir.
- Melakukan pelatihan keamanan.

Mekanisme keamanan harus dilakukan pada tingkatan berikut tanpa mempengaruhi operasi bisnis sehari-hari:

- Keamanan internet (*Web, WAN*)
- Keamanan jaringan internal (*Intranet, LAN*).
- Keamanan sistem operasi
- Keamanan basisdata.
- Keamanan obyek skema.
- Keamanan aplikasi pengguna.
- Keamanan PC (*Log in Windows NT*).

### Referensi:

CS444 Database Management

<http://pheatt.emporia.edu/courses/2004/cs444f04/Lectures.htm>

D. Brent Chapman & Elizabeth D. Zwicky, *"Building Internet Firewalls"*, ISBN 1-51592-124-0, 517 pages. First Edition, Penerbit Oreilly November 1995.

R.K. Stephens dan R.R.Plew., *"Database Design"*, Sams Publishing, 2000.

Semoga **materi** ini bermanfaat bagi kita, dan bisa menjadi acuan materikuliah untuk perancangan basisdata khususnya terhadap keamanan perancangan basisdata.

Terimakasih, Tuhan Memberkati...