KEAMANAN SISTEM DATABASE

Keamanan database adalah suatu cara untuk melindungi database dari ancaman, baik dalam bentuk kesengajaan atau pun bukan.

Ancaman adalah segala situasi atau kejadian baik secara sengaja maupun tidak yang bersifat merugikan dan mempengaruhi system serta secara konsekuensi terhadap perusahaan/organisasi yang memiliki system database.

Keamanan database tidak hanya berkenaan dengan data yang ada pada database saja, tetapi juga meliputi bagian lain dari system database, yang tentunya dapat mempengaruhi database tersebut. Hal ini berarti keamanan database mencakup perangkat keras, perangkat lunak, orang dan data.

Agar memiliki suatu keamanan yang efektif dibutuhkan kontrol yang tepat. Seseorang yang mempunyai hak untuk mengontrol dan mengatur database biasanya disebut Administrator database. Seorang administratorlah yang memegang peranan penting pada suatu system database, oleh karena itu administrator harus mempunyai kemampuan dan pengetahuan yang cukup agar dapat mengatur suatu system databa

Keamanan merupakan suatu proteksi terhadap pengrusakan data dan pemakaian data oleh pemakai yang tidak punya kewenangan.

System yang aman memastikan kerahasian data yang terdapat didalamnya. Beberapa aspek keamanan yaitu :

- Mambatasi akses ke data dan servis
- Melakukan autentifikasi pada user
- Memonitor aktivitas-aktivitas yang mencurigakan

Keamanan database dapat dikelompokan sebagai berikut :

- Pencurian dan penipuan.

  Pencurian dan penipuan database tidak hanya mempengaruhi lingkungan database tetapi juga seluruh perusahaan/organisasi. Keadaan ini dilakukan oleh orang, dimana seseorang ingin melakukan pencurian data atau manipulasi data, seperti saldo rekening,transaksi,transfer dan lain-lain. Untuk itu fokus harus dilakukan

pada kekuatan system agar menghindari akses oleh orang yang tidak memiliki kewenangan.

- Hilangnya kerahasiaan dan privasi

  Suatu data dapat memiliki nilai kerahasiaan, karena data tersebut merupakan sumber daya yang strategis pada perusahaan, maka pada kasus ini data tersebut harus diamankan dengan memberikan hak akses pada orang tertentu saja.

- Hilangnya integritas

  Integritas ini berkaitan dengan akurasi dan kebenaran data dalam database, seperti data korup.Hal ini akan secara serius mempengaruhi perusahaan/organisasi.

- Hilangnya ketersediaan

  Hilangnya ketersediaan berarti data, system, keduanya tidak dapat diakses,servis mati, yang tentunya secara serius sangat mempengaruhi perusahaan/organisasi. Saat ini banyak perusahaan yang membutuhkan kemampuan system yang aktif 7 x 24 , 7 hari 1 minggu.

Berdasarkan pengelompokan tersebut, tentunya banyak aspek yang harus kita perhatikan demi terciptanya keamanan database. Bisa saja seseorang mencuri computer kita yang berisi data penting, mungkin juga karyawan yang diberi hak untuk mengakses data melakukan kejahatan dengan menjual informasi tersebut pada pihak lain demi kepentingan pribadi.Hal-hal tersebut memang termasuk kendala keamanan database yang harus mendapat perhatian, tetapi seorang administrator tidak dapat mengawasi kelemahan tersebut. Seorang administrator hanya fokus pada sistem database itu sendiri, dan hal inilah yang akan kita bicarakan.

Tentunya perkembangan teknologi mengharuskan suatu perusahaan untuk mengimplementasikan system database yang bukan hanya aman tetapi juga mudah diakses dan handal, menyala 7x24 jam, 7 hari 1 minggu tanpaoff.

Penyebaran informasi secara global sangat menguntungkan semua pihak. Dengan adanya internet, komunikasi antar cabang, perusahaan, konsumen dan sebagainya semakin mudah. Pemberian informasi mengenai perusahaan kepada masyarakat melalui internet merupakan salah satu strategi komunikasi, marketing, public relation perusahaan

tersebut,adanya transaksi on line yang meningkatkan gaya hidup masyarakat dan lain-lain. Semua itu tidak terlepas dari suatu perkembangan system database dan tentunya membuat keamanan menjadi rentan.

Sangatlah mudah dalam suatu lingkungan database diciptakan suasana yang menakutkan, tanpa kepastian dan keraguan. Sebagai seorang administrator sangat perlu memperhatikan kondisi tersebut. Tentukan resiko yang sebenarnya dan selidiki apa yang dapat dilakukan terhadap kondisi itu. Sebenarnya kebanyakan database terkonfigurasi dalam keadaan yang mudah ditembus, akan tetapi hal ini bukan berarti database tidak dapat dibuat aman sebagaimana mestinya.

Acaman terhadap database

Serangan terhadap database

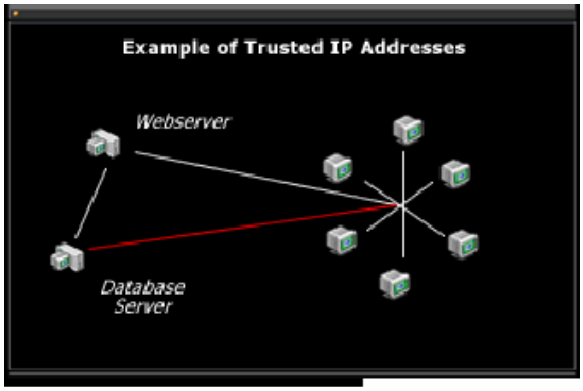
Secara garis besar keamanan database dikategorikan sbb:

- **KeamananServer**

  Perlindungan Server adalah suatu proses pembatasan akses yang sebenarnya pada database dalam server itu sendiri. Menurut Blake Wiedman ini adalah suatu sisi keamanan yang sangat penting dan harus direncanakan secara hati-hati. Ide dasarnya adalah kita tidak dapat mengakses apa yang kita tidak dapat lihat, atau apakah kita ingin database server kita dapat dilihat diseluruh dunia? Database kita bukanlah suatu web server,koneksi yang tidak dikenali tidak diijinkan.

- **Trusted Ip Access**

  Setiap server harus dapat mengkonfigurasikan alamat ip yang diperbolehkan mengakses dirinya. Kita tidak mengijinkan semua orang dapat mengakses server kita sebagaimana kita tidak mengijinkan orang lain memasuki rumah kita tanpa ijin. Jika server melayani suatu web server maka hanya alamat web server itu saja yang dapat mengakses server database tersebut.Jika server database melayani jaringan internal maka hanya alamat jaringanlah yang boleh menghubungi server. Sangat perlu diperhatikan bahwa jangan pernah menggabungkan server database web dengan server database informasi internal perusahaan anda, ini adalah suatu mental yang buruk untuk seorang admin.



Gambar 1.

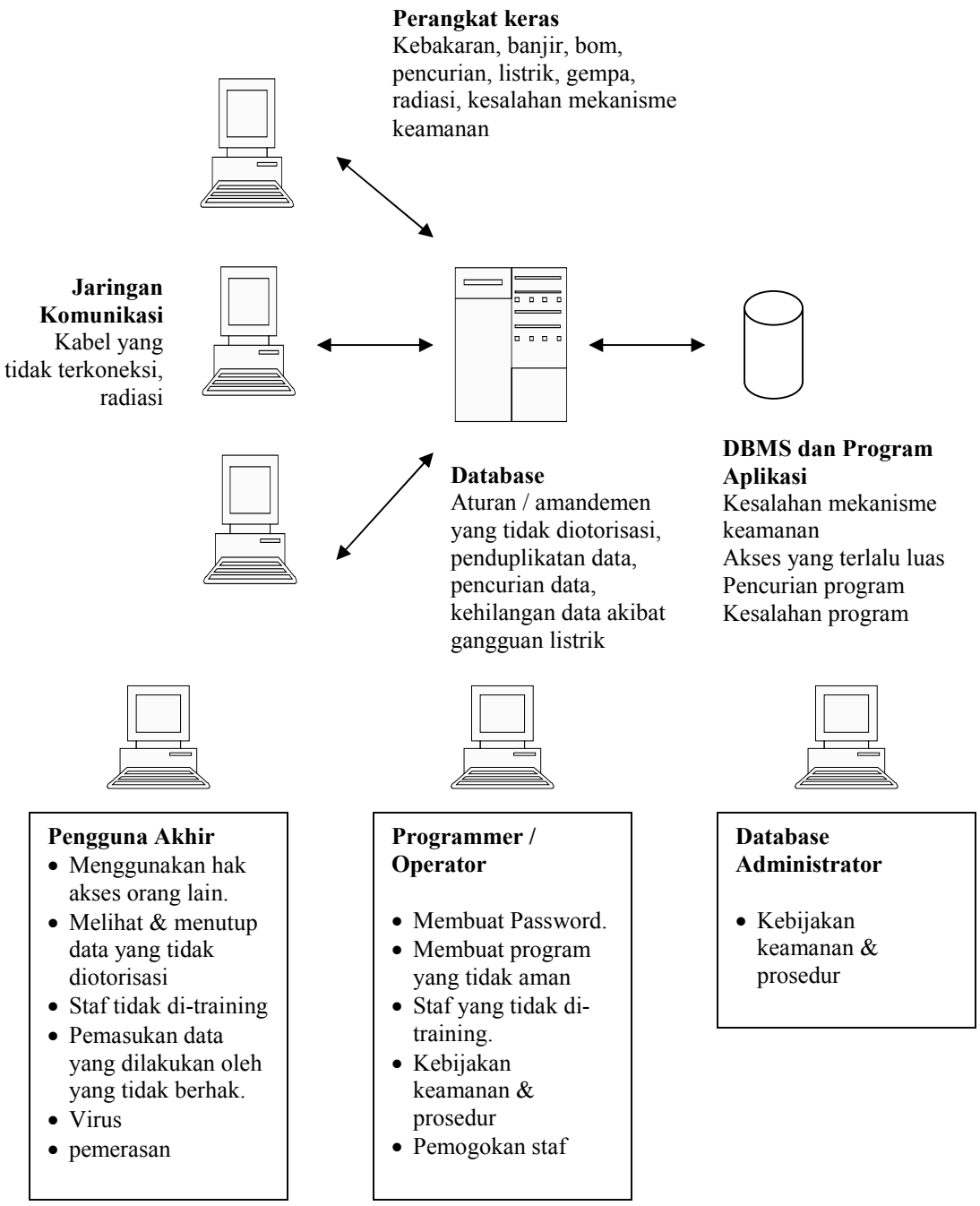
Trusted Ip Accessmerupakan server database terbatas yang hanya akan memberi respon pada Ip yang dikenali saja.

- **Koneksi Database**

  Saat ini semakin banyaknya aplikasi dinamis menjadi sangat menggoda untuk melakukan akses yang cepat bahkan update yang langsung tanpa authentifikasi. Jangan pernah berpikir demikian, ini hanya untuk seorang pemalas.Jika kita ingin mengijinkan pemakai dapat mengubah database melalui web page, pastikan anda memvalidasi semua masukan untuk memastikan bahwa inputan benar, terjamin dan aman.Sebagai contoh, pastikan anda menghilangkan semua code SQL agar tidak dapat dimasukan oleh user.Jika anda seorang admin yang membutuhkan koneksi ODBC,pastikan koneksi yang digunakan unik.

- Kontrol Akses Tabel

  Kontrol akses table ini adalah salah satu bentuk keamanan database yang sering diabaikan,karena cukup sulit penerapannya. Penggunaan control akses table yang benar dibutuhkan kolaborasi antara system administrator dengan pengembang database. Hal inilah yang sulit dilakukan. Pemberian ijin user untuk mengakses informasi dapat membuat informasi terbuka kepada public. Jika seorang user mengakses informasi apakah akan dilihat menggunakan session yang sama? Atau jika table digunakan sebagai referensi system mengapa ia diberikan ijin selain hak membaca saja.

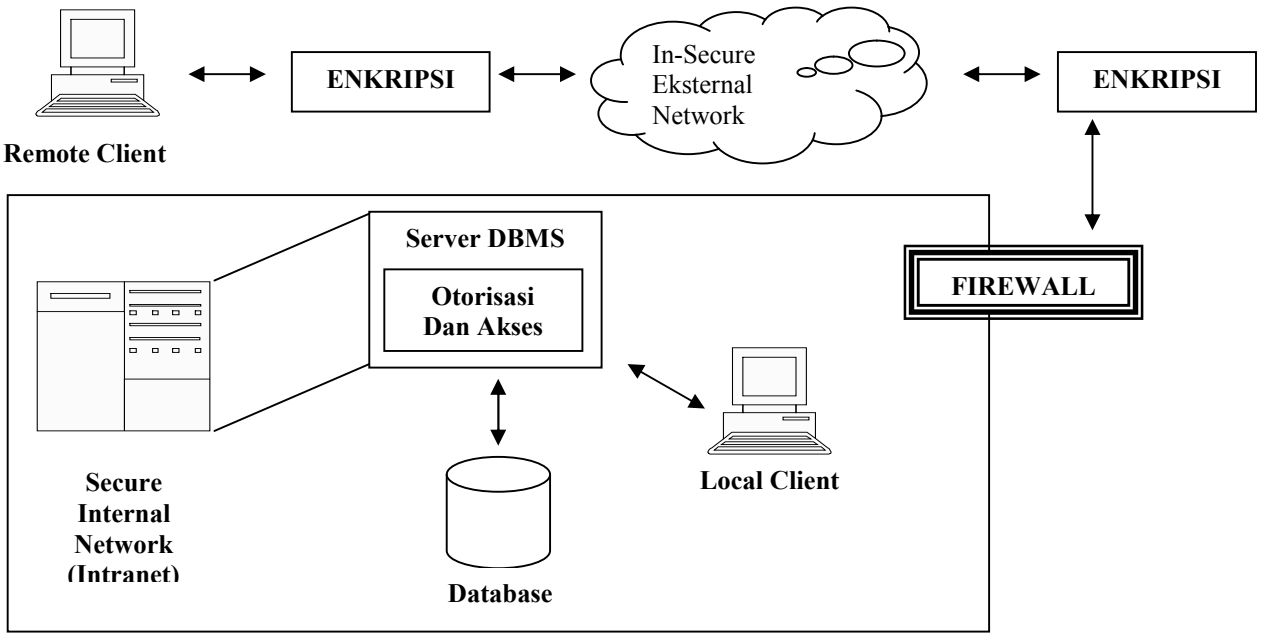

Perangkat keras
Kebakaran, banjir, bom, pencurian, listrik, gempa, radiasi, kesalahan mekanisme keamanan
Jaringan Komunikasi
Kabel yang tidak terkoneksi, radiasi
Database
Aturan / amandemen yang tidak diotorisasi, penduplikatan data, pencurian data, kehilangan data akibat gangguan listrik
DBMS dan Program Aplikasi
Kesalahan mekanisme keamanan
Akses yang terlalu luas
Pencurian program
Kesalahan program
Pengguna Akhir
• Menggunakan hak akses orang lain.
• Melihat & menutup data yang tidak diotorisasi
• Staf tidak di-training
• Pemasukan data yang dilakukan oleh yang tidak berhak.
• Virus
• pemerasan
Programmer / Operator
• Membuat Password.
• Membuat program yang tidak aman
• Staf yang tidak di-training.
• Kebijakan keamanan & prosedur
• Pemogokan staf
Database Administrator
• Kebijakan keamanan & prosedur

Penyalahgunaan Database :

1. Tidak disengaja, jenisnya :
    a. kerusakan selama proses transaksi
    b. anomali yang disebabkan oleh akses database yang konkuren
    c. anomali yang disebabkan oleh pendistribuasian data pada beberapa komputer
    d. logika error yang mengancam kemampuan transaksi untuk mempertahankan konsistensi database.
2. Disengaja, jenisnya :
    a. Pengambilan data / pembacaan data oleh pihak yang tidak berwenang.
    b. Pengubahan data oleh pihak yang tidak berwenang.
    c. Penghapusan data oleh pihak yang tidak berwenang.

Tingkatan Pada Keamanan Database :

1. Fisikal → lokasi-lokasi dimana terdapat sistem komputer haruslah aman secara fisik terhadap serangan perusak.
2. Manusia → wewenang pemakai harus dilakukan dengan berhati-hati untuk mengurangi kemungkinan adanya manipulasi oleh pemakai yang berwenang
3. Sistem Operasi → Kelemahan pada SO ini memungkinkan pengaksesan data oleh pihak tak berwenang, karena hampir seluruh jaringan sistem database menggunakan akses jarak jauh.
4. Sistem Database → Pengaturan hak pemakai yang baik.



Keamanan Data :

1. Otorisasi :

- Pemberian Wewenang atau hak istimewa (priviledge) untuk mengakses sistem atau obyek database
- Kendali otorisasi (=kontrol akses) dapat dibangun pada perangkat lunak dengan 2 fungsi :
- Mengendalikan sistem atau obyek yang dapat diakses
- Mengendalikan bagaimana pengguna menggunakannya
- Sistem administrasi yang bertanggungjawab untuk memberikan hak akses dengan membuat account pengguna.

2. Tabel View :

- Merupakan metode pembatasan bagi pengguna untuk mendapatkan model database yang sesuai dengan kebutuhan perorangan. Metode ini dapat menyembunyikan data yang tidak digunakan atau tidak perlu dilihat oleh pengguna.

- Contoh pada Database relasional, untuk pengamanan dilakukan beberapa level :
    1. Relasi → pengguna diperbolehkan atau tidak diperbolehkan mengakses langsung suatu relasi
    2. View → pengguna diperbolehkan atau tidak diperbolehkan mengakses data yang terapat pada view
    3. Read Authorization → pengguna diperbolehkan membaca data, tetapi tidak dapat memodifikasi.
    4. Insert Authorization → pengguna diperbolehkan menambah data baru, tetapi tidak dapat memodifikasi data yang sudah ada.
    5. Update Authorization → pengguna diperbolehkan memodifikasi data, tetapi tidak dapat menghapus data.
    6. Delete Authorization → pengguna diperbolehkan menghapus data.

- Untuk Modifikasi data terdapat otorisasi tambahan :
    1. Index Authorization → pengguna diperbolehkan membuat dan menghapus index data.
    2. Resource Authorization → pengguna diperbolehkan membuat relasi-relasi baru.
    3. Alteration Authorization → pengguna diperbolehkan menambah/menghapus atribut suatu relasi.
    4. Drop Authorization → pengguna diperbolehkan menghapus relasi yang sudah ada.

- Contoh perintah menggunakan SQL :

  GRANT : memberikan wewenang kepada pemakai

  Syntax : GRANT <priviledge list> ON <nama relasi/view> TO <pemakai>

  Contoh :

  GRANT SELECT ON S TO BUDI

  GRANT SELECT,UPDATE (STATUS,KOTA) ON S TO ALI,BUDI

  REVOKE : mencabut wewenang yang dimiliki oleh pemakai

  Syntax : REVOKE <priviledge list> ON <nama relasi/view> FROM <pemakai>

Contoh :

REVOKE SELECT ON S TO BUDI

REVOKE SELECT,UPDATE (STATUS,KOTA) ON S TO ALI,BUDI

Priviledge list : READ, INSERT, DROP, DELETE, INEX, ALTERATION, RESOURCE

3. Backup data dan recovery :

Backup : proses secara periodik untuk mebuat duplikat ari database dan melakukan logging file (atau program) ke media penyimpanan eksternal.

Jurnaling : proses menyimpan dan mengatur log file dari semua perubahan yang dibuat di database untuk proses recovery yang efektif jika terjadi kesalahan.

Isi Jurnal :

- Record transaksi
    1. Identifikasi dari record
    2. Tipe record jurnal (transaksi start, insert, update, delete, abort, commit)
    3. Item data sebelum perubahan (operasi update dan delete)
    4. Item data setelah perubahan (operasi insert dan update)
    5. Informasi manajemen jurnal (misal : pointer sebelum dan record jurnal selanjutnya untuk semua transaksi
- Record checkpoint : suatu informasi pada jurnal untuk memulihkan database dari kegagalan, kalau sekedar redo, akan sulit penyimpanan sejauh mana jurnal untuk mencarinya kembali, maka untuk membatasi pencarian menggunakan teknik ini.

Recovery : merupakan upaya uantuk mengembalikan basis data ke keadaaan yang dianggap benar setelah terjadinya suatu kegagalan.

3. Jenis Pemulihan :

1. Pemulihan terhadap kegagalan transaksi : Kesatuan prosedur alam program yang dapat mengubah / memperbarui data pada sejumlah tabel.
2. Pemulihan terhadap kegagalan media : Pemulihan karena kegagalan media dengan cara mengambil atau memuat kembali salinan basis data (backup)

3. Pemulihan terhadap kegagalan sistem : Karena gangguan sistem, hang, listrik terputus alirannya.

Fasilitas pemulihan pada DBMS :

1. Mekanisme backup secara periodik
2. fasilitas logging dengan membuat track pada tempatnya saat transaksi berlangsung dan pada saat database berubah.
3. fasilitas checkpoint, melakukan update database yang terbaru.
4. manager pemulihan, memperbolehkan sistem untuk menyimpan ulang database menjadi lebih konsisten setelah terjadinya kesalahan.

Teknik Pemulihan :

1. **defered upate / perubahan yang ditunda** : perubahan pada DB tidak akan berlangsung sampai transaksi ada pada poin disetujui (COMMIT). Jika terjadi kegagalan maka tidak akan terjadi perubahan, tetapi diperlukan operasi redo untuk mencegah akibat dari kegagalan tersebut.
2. **Immediate Upadate / perubahan langsung** : perubahan pada DB akan segera tanpa harus menunggu sebuah transaksi tersebut disetujui. Jika terjadi kegagalan diperlukan operasi UNDO untuk melihat apakah ada transaksi yang telah disetujui sebelum terjadi kegagalan.
3. **Shadow Paging** : menggunakan page bayangan imana paa prosesnya terdiri dari 2 tabel yang sama, yang satu menjadi tabel transaksi dan yang lain digunakan sebagai cadangan. Ketika transaksi mulai berlangsung kedua tabel ini sama dan selama berlangsung tabel transaksi yang menyimpan semua perubahan ke database, tabel bayangan akan digunakan jika terjadi kesalahan. Keuntungannya adalah tidak membutuhkan REDO atau UNDO, kelemahannya membuat terjadinya fragmentasi.

4. Kesatuan data dan Enkripsi :

Enkripsi : keamanan data

- Integritas :metode pemeriksaan dan validasi data (metode integrity constrain), yaitu berisi aturan-aturan atau batasan-batasan untuk tujuan terlaksananya integritas data.
- Konkuren : mekanisme untuk menjamin bahwa transaksi yang konkuren pada database multi user tidak saling menganggu operasinya masing-masing. Adanya penjadwalan proses yang akurat (time stamping).

Fasilitas Keamanan Database

Keamanan database tersedia pada versi Educator ke atas. Keamanan database diatur oleh Properti Database. Berikut ini adalah properti database yang digunakan untuk keamanan database BOCSoft eQuestion.

Properti	Keterangan
1. Publikasi	Apakah database dipublikasikan? Database yang telah dipublikasikan tidak bisa dipublikasikan ulang. Proses publikasi adalah mempublikasikan database untuk konsumsi publik. Proses ini meliputi pengaturan properti lain: Proteksi; Hanya Baca; Dapat Dibaca eQuestion Lain; dan Dapat Diimpor.
2. Proteksi	Jika database diproteksi, setiap menggunakan database, pengguna akan dimintai password/kata kunci sebagai pengaman database. Password ditentukan oleh pembuat database.
3. Hanya Baca (Read-Only)	Data dalam database yang "Hanya Baca" tidak bisa ditambah, diedit, atau dihapus.

4. Dapat Dibaca eQuestion Lain	Jika properti ini diset "Tidak" maka database hanya bisa dibaca oleh BOCSoft eReader dan tidak bisa dibaca oleh BOCSoft eQuestion lain.
5. Dapat Diimpor	Jika properti ini diset "Ya" maka data dari database eQuestion bisa digabungkan dengan database eQuestion lain dengan versi yang sama.

Tingkatan Pada Keamanan Database

1. Fisikal ; lokasi-lokasi dimana terdapat sistem komputer haruslah aman secara fisik terhadap serangan perusak.
2. Manusia ; wewenang pemakai harus dilakukan dengan berhati-hati untuk mengurangi kemungkinan adanya manipulasi oleh pemakai yang berwenang
3. Sistem Operasi ; Kelemahan pada SO ini memungkinkan pengaksesan data oleh pihak tak berwenang, karena hampir seluruh jaringan sistem database menggunakan akses jarak jauh.
4. Sistem Database ; Pengaturan hak pemakai yang baik.

Enkripsi Untuk Keamanan Database

Salah satu hal yang penting dalam komunikasi menggunakan computer untuk menjamin kerahasian data adalah enkripsi. Enkripsi dalah sebuah proses yang melakukan perubahan sebuah kode dari yang bisa dimengerti menjadi sebuah kode yang tidak bisa dimengerti (tidak terbaca). Enkripsi dapat diartikan sebagai kode atau chiper. Sebuah sistem pengkodean menggunakan suatu table atau kamus yang telah didefinisikan untuk mengganti kata dari informasi atau yang merupakan bagian dari informasi yang dikirim. Sebuah chiper menggunakan suatu algoritma yang dapat mengkodekan semua aliran data (stream) bit dari sebuah pesan menjadi cryptogram yang tidak dimengerti (unitelligible). Karena teknik cipher merupakan suatu sistem yang telah siap untuk di automasi, maka teknik ini digunakan dalam sistem keamanan komputer dan network.

Pada bagian selanjutnya kita akan membahas berbagai macam teknik enkripsi yang biasa digunakan dalam sistem security dari sistem komputer dan network.

A. Enkripsi Konvensional.

Proses enkripsi ini dapat digambarkan sebagai berikut :

Plain teks -> Algoritma Enkripsi -> Cipher teks ->Algoritma Dekrispsi -> Plain teks

User A | | User B

|————————————-Kunci (Key) ————————————|

Gambar 1

Informasi asal yang dapat di mengerti di simbolkan oleh Plain teks, yang kemudian oleh algoritma Enkripsi diterjemahkan menjadi informasi yang tidak dapat untuk dimengerti yang disimbolkan dengan cipher teks. Proses enkripsi terdiri dari dua yaitu algoritma dan kunci. Kunci biasanya merupakan suatu string bit yang pendek yang mengontrol algoritma. Algoritma enkripsi akan menghasilkan hasil yang berbeda tergantung pada kunci yang digunakan. Mengubah kunci dari enkripsi akan mengubah output dari algortima enkripsi.

Sekali cipher teks telah dihasilkan, kemudian ditransmisikan. Pada bagian penerima selanjutnya cipher teks yang diterima diubah kembali ke plain teks dengan algoritma dan dan kunci yang sama.

Keamanan dari enkripsi konvensional bergantung pada beberapa factor. Pertama algoritma enkripsi harus cukup kuat sehingga menjadikan sangat sulit untuk mendekripsi cipher teks dengan dasar cipher teks tersebut. Lebih jauh dari itu keamanan dari algoritma enkripsi konvensional bergantung pada kerahasian dari kuncinya bukan algoritmanya. Yaitu dengan asumsi bahwa adalah sangat tidak praktis untuk mendekripsikan informasi dengan dasar cipher teks dan pengetahuan tentang algoritma diskripsi / enkripsi. Atau dengan kata lain, kita tidak perlu menjaga kerahasiaan dari algoritma tetapi cukup dengan kerahasiaan kuncinya.

Manfaat dari konvensional enkripsi algoritma adalah kemudahan dalam penggunaan secara luas. Dengan kenyataan bahwa algoritma ini tidak perlu dijaga kerahasiaannya dengan maksud bahwa pembuat dapat dan mampu membuat suatu implementasi dalam bentuk chip dengan harga yang murah. Chips ini dapat tersedia secara luas dan disediakan pula untuk beberapa jenis produk. Dengan penggunaan dari enkripsi konvensional, prinsip keamanan adalah menjadi menjaga keamanan dari kunci.

Model enkripsi yang digunakan secara luas adalah model yang didasarkan pada data encrytion standard (DES), yang diambil oleh Biro standart nasional US pada tahun 1977. Untuk DES data di enkripsi dalam 64 bit block dengan menggunakan 56 bit kunci. Dengan menggunakan kunci ini, 64 data input diubah dengan suatu urutan dari metode menjadi 64 bit output. Proses yang yang sama dengan kunci yang sama digunakan untuk mengubah kembali enkripsi.

B. Enkripsi Public-Key

Salah satu yang menjadi kesulitan utama dari enkripsi konvensional adalah perlunya untuk mendistribusikan kunci yang digunakan dalam keadaan aman. Sebuah cara yang tepat telah diketemukan untuk mengatasi kelemahan ini dengan suatu model enkripsi yang secara mengejutkan tidak memerlukan sebuah kunci untuk didistribusikan. Metode ini dikenal dengan nama enkripsi public-key dan pertama kali diperkenalkan pada tahun 1976.

Plain teks -> Algoritma Enkripsi -> Cipher teks -> Algoritma Dekrispsi -> Plain teks

User A | | User B

Private Key B —-|

|————————-Kunci (Key) ————————|

Gambar 2

Algoritma tersebut seperti yang digambarkan pada gambar diatas. Untuk enkripsi konvensional, kunci yang digunakan pada prosen enkripsi dan dekripsi adalah sama. Tetapi ini bukanlah kondisi sesungguhnya yang diperlukan. Namun adalah dimungkinkan untuk membangun suatu algoritma yang menggunakan satu kunci untuk enkripsi dan pasangannya, kunci yang berbeda, untuk dekripsi. Lebih jauh lagi adalah mungkin untuk

menciptakan suatu algoritma yang mana pengetahuan tentang algoritma enkripsi ditambah kunci enkripsi tidak cukup untuk menentukan kunci dekrispi. Sehingga teknik berikut ini akan dapat dilakukan :

1. Masing – masing dari sistem dalam network akan menciptakan sepasang kunci yang digunakan untuk enkripsi dan dekripsi dari informasi yang diterima.
2. Masing – masing dari sistem akan menerbitkan kunci enkripsinya (public key) dengan memasang dalam register umum atau file, sedang pasangannya tetap dijaga sebagai kunci pribadi (private key).
3. Jika A ingin mengisim pesan kepada B, maka A akan mengenkripsi pesannya dengan kunci publik dari B.
4. Ketika B menerima pesan dari A maka B akan menggunakan kunci privatenya untuk mendeskripsi pesan dari A.

Seperti yang kita lihat, public-key memecahkan masalah pendistribusian karena tidak diperlukan suatu kunci untuk didistribusikan. Semua partisipan mempunyai akses ke kunci publik (public key) dan kunci pribadi dihasilkan secara lokal oleh setiap partisipan sehingga tidak perlu untuk didistribusikan. Selama sistem mengontrol masing – masing private key dengan baik maka komunikasi menjadi komunikasi yang aman. Setiap sistem mengubah private key pasangannya public key akan menggantikan public key yang lama. Yang menjadi kelemahan dari metode enkripsi publik key adalah jika dibandingkan dengan metode enkripsi konvensional algoritma enkripsi ini mempunyai algoritma yang lebih komplek. Sehingga untuk perbandingan ukuran dan harga dari hardware, metode publik key akan menghasilkan performance yang lebih rendah. Tabel berikut ini akan memperlihatkan berbagai aspek penting dari enkripsi konvensional dan public key.

- **Enkripsi Konvensional**

  Yang dibutuhkan untuk bekerja :

1. Algoritma yang sama dengan kunci yang sama dapat digunakan untuk proses dekripsi–enkripsi. Pengirim dan penerima harus membagi algoritma dan kunci yang sama.

Yang dibutuhkan untuk keamanan :

1. Kunci harus dirahasiakan.
2. Adalah tidak mungkin atau sangat tidak praktis untuk menerjemahkan informasi yang telah dienkripsi.
3. Pengetahuan tentang algoritma dan sample dari kata yang terenkripsi tidak mencukupi untu menentukan kunc.

- **Enkripsi Public Key**

  Yang dibutuhkan untuk bekerja :

  1. Algoritma yang digunakan untuk enkripsi dan dekripsi dengan sepasang kunci, satu untuk enkripsi satu untuk dekripsi.
  2. Pengirim dan penerima harus mempunyai sepasang kunci yang cocok.

Yang dibutuhkan untuk keamanan :

1. Salah satu dari kunci harus dirahasiakan.
2. Adalah tidak mungkin atau sangat tidak praktis untuk menerjemahkan informasi yang telah dienkripsi.
3. Pengetahuan tentang algoritma dan sample dari kata yang terenkripsi tidak mencukupi untu menentukan kunci.