ASPECTOS OPERACIONAIS: SEGURANÇA

Valéria Times

Introdução a Segurança em BD

- Aspectos a serem considerados
  - Éticos e legais relacionados ao direito de acesso a certas informações
  - Políticos, a nível governamental, institucional ou corporativo, relativo às informações que não devem estar publicamente disponíveis
  - Relativos ao sistema tais como os níveis nos quais as funções de segurança devem ser manipuladas

Introdução a Segurança em BD

- Aspectos a serem considerados (Cont.)
  - Necessidade, em algumas organizações, de identificar vários níveis de segurança e categorizar os dados e os usuários de acordo

→ **Sub-sistema de Segurança e Autorização no SGBD**

Introdução a Segurança em BD

- Objetivos para uma Aplicação de BD Segura
  - **Sigilo:** Usuários não devem acessar dados aos quais não têm permissão
    - Ex: Um estudante não pode ver notas de outros estudantes
  - **Integridade:** Usuários não devem modificar dados sem permissão
    - Ex: Somente professores podem dar notas
  - **Disponibilidade:** Usuários devem poder modificar e acessar dados aos quais tenham permissão

Introdução a Segurança no BD

- Responsabilidades do DBA
  - Dar privilégios a usuários
  - Classificar usuários e dados de acordo com a política da organização

↳ **Conta privilegiada (root, super-usuário)**

- Ações do DBA
  - Abrir contas

    Conta e password permitindo o acesso ao SGBD
  - Atribuir privilégios

    Atribuir certos privilégios a certas contas

Introdução a Segurança no BD

- Ações do DBA (Cont.)
  - Retirar privilégios

    Cancelar privilégios atribuídos anteriormente
  - Definir níveis de segurança

    Define as contas do usuário nos níveis de segurança apropriado

Responsável pela Segurança Geral do Sistema de BD

Introdução a Segurança em BD

- Uma política de segurança especifica quem tem autorização e para que.
- Um mecanismo de segurança nos permite forçar o uso de uma política de segurança.
- Dois mecanismos no nível do SGBD são:
  - Mecanismo Arbitrário
    Usado para dar privilégios a usuários para acessar arquivos, registros ou campos de dados
  - Mecanismo Obrigatório
    Usado para classificar dados e usuários em diversas classes de segurança

Introdução a Segurança em BD

- Outras funcionalidades
  - Controle de acesso: Prevenir contra pessoas não autorizadas
  - Segurança em BD Estatísticos: Garantir que informações individuais não possam ser acessadas
  - Dados Encriptados: Proteger dados sensíveis que estão sendo transmitidos via rede ou para prover proteção adicional a partes do BD

⬇

Codificado por algum tipo de algoritmo

Controle de Acesso

- Proteção de acesso
  Número de conta e password são verificados quando do login
- Controle de login
  Seqüência de operações feitas por um dado usuário - do login ao logout

⬇

- Modificar o arquivo de log
  Incluir a identificação da conta do usuário e a identificação do terminal onde está logado

Qualquer Problema

↳ **AUDITORIA DO BD**

Controle de Acesso Arbitrário

- Baseado em Privilégios
  Consiste no conceito de direito de acesso ou privilégio a objetos (tabelas e visões), e mecanismos para conceder e revogar privilégios de usuários.
- Método Típico
  - Dar (GRANT)
  - Retirar (REVOKE)

  } **PRIVILÉGIOS**
- Criador de um objeto (tabela ou visão) automaticamente tem todos os privilégios sobre o que criou.

Controle de Acesso Arbitrário

- SGBD
  - Mantém dados sobre quem recebeu e perdeu privilégios (tabela de privilégios)
    - Provedor do privilégio
    - Receptor do privilégio
    - Tipo de privilégio, incluindo o ID do objeto
    - Indicação se propagação foi concedida
  - Insere automaticamente uma entrada nesta tabela quando um objeto (tabela ou visão) é criado, com o sistema sendo o provedor

Controle de Acesso Arbitrário

- SGBD (Cont.)
  - Garante que somente solicitações feitas por usuários que tenham os privilégios necessários (no momento em que a requisição foi feita) sejam permitidas
  - Permite que apenas o dono possa executar CREATE, ALTER e DROP.
  - Provê o comando GRANT

GRANT privilégios ON objeto TO usuários
[WITH GRANT OPTION]

Controle de Acesso Arbitrário

- Os seguintes privilégios podem ser especificados:
  - SELECT: Pode ler todas as colunas (incluindo aquelas adicionadas posteriormente via o comando ALTER TABLE)
  - INSERT (nome-col): Pode inserir tuplas com valores não nulos ou não default na coluna especificada. Parecido com UPDATE (nome-col).
    - INSERT: Significa o mesmo direito só que para todas colunas. Parecido com UPDATE.
  - DELETE: Pode remover tuplas
  - REFERENCES (nome-col): Pode definir chaves estrangeiras (em outras tabelas) que se referem à coluna especificada

Controle de Acesso Arbitrário

- Propagação de privilégios

Quando se dá privilégios a uma conta, pode-se dar ou não a ela a opção de dar privilégios a outras, através da

GRANT OPTION

- Propagação de privilégios sem o conhecimento do proprietário
- Se um usuário tem privilégios com GRANT OPTION, então ele pode passar seus privilégios para outros (com ou sem GRANT OPTION).

Controle de Acesso Arbitrário

- Técnicas para limitar a propagação de privilégios
  - Propagação horizontal

    Limitada a i significa que uma conta à qual foi dado o privilégio com GRANT OPTION pode dar privilégios a no máximo i outras contas
  - Propagação vertical

    Associada a j significa que uma conta tem o privilégio GRANT OPTION mas só pode dar privilégios a outras contas com propagação vertical < j

Controle de Acesso Arbitrário

- Exemplos
  - Supor que Ana criou as tabelas

EMPREGADO(nome, id, data_nasc, end, sexo, salário, num_dep)

DEPARTAMENTO(num_depto, nome, id_gerente)

  - Ana é a proprietária e portanto pode dar privilégios sobre essas tabelas
    - grant insert, delete on EMPREGADO, DEPARTAMENTO to Ida;

      Ida não pode dar privilégios a outras contas
    - grant select on EMPREGADO, DEPARTAMENTO to Bia with grant option;

      Bia pode dar privilégios a outras contas
      - grant select on EMPREGADO to Carlos;

Controle de Acesso Arbitrário

- Exemplos (Cont.)
  - grant insert, select on EMPREGADO to Oscar;

    Oscar pode consultar ou inserir tuplas em Empregado
  - grant delete on EMPREGADO to Lia with grant option;

    Lia pode apagar tuplas e autorizar outros para tal.
  - grant update(salário) on EMPREGADO to João;

    João pode modificar apenas o campo salário em Empregado

Controle de Acesso Arbitrário

- Retirando privilégios
  - Em alguns casos é desejável dar privilégios temporariamente (ex.: uma dada tarefa).
  - Comando REVOKE

REVOKE [GRANT OPTION FOR] privilégios ON objeto FROM usuários [RESTRICT / CASCADE]

  - Pode ser usado para remover um privilégio ou apenas a GRANT OPTION de um privilégio

Controle de Acesso Arbitrário

- Retirando privilégios(Cont.)
  - É possível para uma conta receber um certo privilégio de duas ou mais fontes, só perdendo o privilégio quando todas o retirarem
  - Quando um privilégio é retirado de X (cascade), ele também é removido de usuários que tenham recebido este privilégio *apenas* de X

Controle de Acesso Arbitrário

- Exemplos
  - Supor que João criou as tabelas EMPREGADO e DEPARTAMENTO
    - grant select on EMPREGADO to Ana with grant option; (executado por João)
    - grant select on EMPREGADO to Bia with grant option; (executado por Ana)
    - Revoke select on EMPREGADO from Ana cascade (executado por João)

    Bia e Ana perdem os privilégios automaticamente.

    Bia teria mantido o privilégio se o tivesse recebido a partir de outra fonte (João).

Controle de Acesso Arbitrário

- Especificando autorizações através de visões

  Mecanismo de autorização importante por permitir o acesso à parte (sub-conjunto de atributos ou tuplas) de uma relação
  - Exemplo:
    - André é o proprietário da relação R
    - André cria uma visão V com alguns atributos de R
    - André dá a Bruno o privilégio SELECT em V

Controle de Acesso Arbitrário

- Visões e Segurança
  - Visões podem ser usadas para exibir dados necessários, enquanto escondem detalhes em tabelas subjacentes
    - Dado AlunosInformática, mas não Alunos ou Matrícula, pode-se obter alunos matriculados, mas não os *nomes* das disciplinas matriculadas
  - Criador de uma visão tem um determinado privilégio sobre a visão se tem este privilégio em todas tabelas usadas pela visão

Controle de Acesso Arbitrário

- Visões e Segurança(Cont.)
  - Juntamente com comandos GRANT / REVOKE, visões são ferramentas poderosas para o controle de acesso.

Controle de Acesso Arbitrário

- GRANT/REVOKE em Visões
  - Para criar uma visão, o usuário deve ter o privilégio SELECT em todas as relações envolvidas na definição da visão
  - Se o criador de uma visão perde o privilégio de SELECT em uma tabela usada pela visão, então a visão é apagada
  - Se um usuário repassa um privilégio com GRANT OPTION e ele perde o privilégio de acesso a uma tabela usada na visão, ele e outros usuários que haviam ganho o privilégio por meio dele perdem o privilégio da visão

Controle de Acesso Arbitrário

- GRANT/REVOKE em Visões

```
create  view  DEPTO5  as
 select  nome,  data_nasc,  end
 from    EMPREGADO
 where  num_dep = 5;
```

É criada uma visão de EMPREGADO para Ana

grant select on DEPTO5 to Ana with grant option;

É dado privilégio a Ana com propagação

grant uptade(salário) on EMPREGADO to Artur;

É dado privilégio a Artur de atualizar apenas salário

Controle de Acesso Arbitrário

- Controle de Acesso baseado em Papéis
  - No SQL-92, privilégios eram dados a *user ids*, que podiam denotar um usuário ou um grupo de usuários
  - No SQL-99 (e em muitos sistemas atuais), privilégios são dados para papéis
    - Papéis podem então ser concedidos para outros usuários ou outros papéis.
    - Reflete como as organizações trabalham
    - Ilustra como padrões freqüentemente evoluem para se adequar a padrões usados em sistemas reais.

Controle de Acesso Arbitrário

- Segurança a nível de campos
  - Pode-se criar uma visão que retorna apenas um campo de uma tupla? E então dar acesso a esta visão?
  - Permite controle com granularidade *arbitrária*, entretanto:
    - É difícil de especificar, apesar disto poder ser minimizado com uma GUI.
    - Desempenho pode tornar-se inaceitável, se for preciso usar granularidade a nível de campos freqüentemente (muitas criações de views e look-ups).

Controle de Acesso Obrigatório

- Baseado em Segurança Multinível

  Mecanismo de segurança que classifica dados e usuários baseado em classes de segurança
- Baseado em políticas globais dos sistema que não podem ser alteradas por certos usuários.
  - Cada objeto do BD é associado a uma classe de segurança
  - Cada usuário ou programa é associado a um passe para uma classe de segurança
  - Regras baseadas em classes de segurança e direitos governam quem pode ler/escrever quais objetos

Controle de Acesso Obrigatório

- Pouco utilizado em SGBDs comerciais
  - Apenas algumas versões de SGBD usam-no
  - Usados para aplicações especializadas (ex. militar)
- Surgiu com o intuito de resolver alguns dos problemas existentes (ex. cavalo de tróia) no acesso aleatório

Controle de Acesso Obrigatório

- Por que Usá-lo?
  - Davi cria tabela T e dá privilégio de INSERT para João (que não sabe nada sobre as más intenções de Davi)
  - Davi modifica o código de alguma aplicação usada por João para adicionalmente escrever dados secretos na tabela T
  - Agora, Davi pode ver a informação sigilosa
- A modificação do código está além do controle do SGBD, mas pode-se prevenir tal problema

Controle de Acesso Obrigatório

- Modelo Bell-LaPadula
  - Objetos (ex. tabelas, visões, tuplas)
  - Sujeitos (ex. usuários, programas)
  - Classes de Segurança:
    - Top Secret (TS), Secret(S), Confidencial (C), Não Classificada (U): TS > S > C > U
  - Cada objeto e sujeito é ligado a uma classe
    - Sujeito S pode ler Objeto O só se classe(S) >= classe (O)
    - Sujeito S pode escrever no Objeto O só se classe(S) <= classe (O)

Controle de Acesso Obrigatório

- A idéia é assegurar que a informação nunca vai de um nível alto para um nível mais baixo
- Como resolver o problema do cavalo de tróia?
  - Se Davi tem classe C, João tem classe S e a tabela secreta tem classe S:
    - Tabela T tem classe de Davi: C
    - A aplicação de João tem sua classe: S
    - Então, o programa não poderá escrever na tabela T
- As regras impostas pelo Obrigatório complementam o Arbitrário

Controle de Acesso Obrigatório

- Classes de segurança usuais
  - Muito secreta (TS)
  - Secreta (S)
  - Confidencial (C)
  - Não classificada (U)

  Onde TS > S > C > U

- Noções de segurança multinível no relacional
  - A cada atributo é associado um atributo de classificação e cada valor de atributo com sua classificação de segurança correspondente

Controle de Acesso Obrigatório

- A cada tupla é adicionado um atributo de classificação de tuplas para classificar a tupla como um todo

  R(A1, C1, A2, C2, ..., An, Cn, TC)

NOME	SALÁRIO	PERFORMANCE	TC
Smith U	4000 C	Fraco S	S
Brown C	8000 S	Bom C	S

Controle de Acesso Obrigatório

- Relações de Multiníveis

BARCO	NOME	COR	TC
101	Diogo	Azul	S
102	Marina	Rosa	C

- Usuários com:
  - classe S e TS conseguem ver ambas as linhas
  - classe C vê somente a segunda linha
  - classe U não vê nenhuma

Controle de Acesso Obrigatório

- Relações de Multiníveis (Cont.)
- Se um usuário C tenta inserir <101,Exemplo,Branco,C >:
  - Permitir a inserção viola a restrição de chave
  - Não deixar a inserção, informa ao usuário que há outro objeto com chave 101 que tem classe de segurança > C !
  - Problema é resolvido considerando a classe como parte da chave.

Controle de Acesso Obrigatório

- Chave Aparente
  Equivalente à chave primária
- Filtragem
  Armazenar tuplas com nível de classificação maior e produzir tuplas correspondentes com nível menor
- Poli-instanciação
  Armazenar duas ou mais tuplas em diferentes níveis de classificação com a mesma chave