UNIVERSIDADE FEDERAL DE UBERLÂNDIA
FACULDADE DE ENGENHARIA ELÉTRICA

TAREFA 03
BANCO DE DADOS – AUTENTICAÇÃO DE USUÁRIOS E SEGURANÇA

08/10/2020

VICTOR GABRIEL CASTÃO DA CRUZ – 11911ECP004

UBERLÂNDIA
2020

1 - INTRODUÇÃO

Com base nas aulas e orientação do professor, a tarefa consistia na criação de uma lista de usuários, os quais atuariam sobre o schema utilizado durante as aulas (“empresax”).

Tais usuários estariam contidos em três grupos, denominados “Gerente”, “Analista” e “Estagiário”. Cada um desses grupos possuiria características próprias, podendo realizar determinadas tarefas em tabelas previamente especificadas, bem como executar funções e até mesmo dar garantias a outros usuários, tudo isso dependendo do grau em que o grupo se encontrasse na hierarquia da empresa.

2 - DESENVOLVIMENTO

2.1 - CRIAÇÃO DOS GRUPOS

Considerando os grupos mencionados anteriormente, estes seriam adicionados ao banco de dados via linhas de comando. Além disso, para exemplificar o que os membros dos grupos poderiam ou não executar, criaram-se, também, três usuários (“Mário”, “Joaquim” e “Paulo”), que seriam adicionados, futuramente, aos grupos criados:

```
create group gerente;
create group analista;
create group estagiario;
create user mario with password 'mario123';
create user joaquim with password 'joaquim123';
create user paulo with password 'paulo123';
```

2.2 – DANDO PODERES AO GERENTE

Segundo o enunciado da tarefa, o grupo dos gerentes teria poder para modificar todos os registros de todas as tabelas, bem como utilizar rotinas (stored procedures) e dar direitos a outros usuários.

Além disso, considerando a possibilidade de dados serem incluídos ou removidos das tabelas, bem como outras alterações (como criação de novas tabelas), foi concedido ao gerente, para todas as tabelas, todos os privilégios.

Levando em consideração a possibilidade de dar poderes a outros usuários, todas as garantias dadas ao gerente, na linha de comando, terminavam com “with grant option”, o que permitia a ele dar o mesmo poder que ele tinha à outro:

```
grant all on schema empresax to group gerente with grant option;
grant all privileges on all tables in schema empresax to group gerente with grant option;
grant all privileges on all functions in schema empresax to group gerente with grant option;
```

2.3 – DANDO PODERES AO ANALISTA

Segundo o enunciado da tarefa, o analista tinha a capacidade de modificar registros, porém não poderia acessar rotinas e nem conceder poderes à outros.

Considerando que o analista não poderia ter as mesmas garantias que o gerente (por uma questão de hierarquia), esse grupo poderia apenas inserir, deletar, atualizar e visualizar dados. Entretanto, essa garantia seria dada somente à tabela de funcionários e auditoria (pois sem não seria possível agir sobre a tabela dos funcionários), pois dessa forma, qualquer outra alteração seria de responsabilidade do gerente, que em tese, terá a experiência necessária para procedimentos mais avançados.

Além disso, foi garantido o acesso ao schema utilizado, pois sem isso, a permissão para agir sobre as tabelas seria negada:

```
grant all on schema empresax to group analista;
grant insert, delete, update, select on empresax.tb_empregados, empresax.tb_empregados_auditoria to group analista;
```

2.4 – DANDO PODERES AO ESTAGIÁRIO

Considerando o enunciado da tarefa, o estagiário seria o grupo com menor poder dentro da empresa (algo justificável devido sua pouca experiência). Sendo assim, este só poderia visualizar os dados tabelas.

Semelhante ao analista, a garantia de acesso ao schema também foi dada. Além disso, as únicas tabelas acessíveis seriam as mesmas do analista (pois conceder poderes em outras tabelas para um grupo de menor hierarquia seria incoerente), porém com direito de apenas visualizar:

```
grant all on schema empresax to group estagiario;
```

```
grant select on empresax.tb_empregados, empresax.tb_empregados_auditoria to group estagiario;
```

2.5 – ADICIONANDO MEMBROS

Como dito no início do relatório, haveriam três usuários criados, cada um pertencendo a um dos grupos criados.

Após a configuração dos grupos, Mário ficou com o cargo de gerente, Joaquim com o de analista e Paulo como estagiário:

```
grant gerente to mario;
grant analista to joaquim;
grant estagiario to paulo;
```

2.6 – AGINDO COM OS USUÁRIOS

Para exemplificar os poderes de ação de cada um dos usuários, alguns comandos foram executados com estes, a fim de verificar o que cada um poderia fazer.

Mário, por exemplo, poderia garantir a Paulo o direito de inserir dados, da mesma forma que poderia revogar:

```
grant insert on empresax.tb_empregados to paulo; --Permitido
revoke insert on empresax.tb_empregados from paulo; --Permitido
```

Entretanto, não poderia criar um novo schema:

```
create schema teste; --Negado
```

Joaquim, na condição de analista, poderia cadastrar um novo funcionário:

```
insert into empresax.tb_empregados values ('Dario', 6500.76); --Permitido
```

Porém, não consegue criar um novo usuário:

```
create user convidado; --Negado
```

Paulo, por ser estagiário, consegue visualizar os funcionários cadastrados:

```
select * from empresax.tb_empregados; --Permitido
```

Entretanto, não consegue cadastrar um novo funcionário:

```
insert into empresax.tb_empregados values('Dario', 6500.76); --Negado
```

3 – CONCLUSÃO

Um banco de dados pode apresentar uma elevada complexidade quando for constituído de uma série de tabelas, funções e outros elementos.

Dessa forma, em alguns casos são necessários diversos usuários para administrar tal volume de informações. Entretanto, é necessário também estabelecer uma certa hierarquia de comando entre tais usuários por uma questão de organização e segurança.

Para exemplificar tal necessidade, imagine uma situação onde qualquer usuário tivesse poder para fazer o que bem entender. Com total certeza, em pouco tempo haveriam dados duplicados ou incorretos, sem considerar ainda uma possível ação precipitada que poderia comprometer todo o banco de dados. Em resumo, a chance de problemas seria alta.

Considerando isso, o exemplo desenvolvido criou três diferentes grupos de "administradores" do banco de dados, cada um com seus devidos poderes e limites de atuação.

Através dos exemplos, verificou-se que, dessa forma, algumas ações simples poderiam ser realizadas por todos, enquanto ações cruciais só são concedidas a grupos devidamente capacitados, contribuindo para um banco de dados organizado e com baixa probabilidade de ocorrerem erros.