ESCOLA SUPERIOR DE TECNOLOGIA E GESTÃO
INSTITUTO POLITÉCNICO DE LEIRIA

Sistemas de Bases de Dados

ENGENHARIA INFORMÁTICA 2019/20

2.º ano, 2.º semestre

Ficha de Trabalho n.º 1 – Segurança em Bases de Dados (parte 2)

Objetivos:

- Gerir limites de recursos
- Gerir/consultar o mecanismo de auditoria na base de dados
- Utilizar cifragem/encriptação de dados

Antes de iniciar a resolução desta ficha de trabalho deverá responder às seguintes questões:

1. Diga o entende por:
   a. Perfil;
   b. Auditoria.
   c. Cifragem/Encriptação;

2. Apresente um exemplo do comando que permite realizar cada uma das seguintes ações:
   a. Retirar um privilégio de sistema de um role;
   b. Atribuir um perfil a um utilizador;
   c. Ativar a auditoria para uma determinada ação na base de dados.

3. Apresente um exemplo da aplicação de encriptação a dados de uma tabela.

CASO DE ESTUDO

Uma determinada escola de condução dedica-se ao ensino da condução a membros da comunidade com mais de 18 anos de idade. Para se ser aluno e ter acesso às aulas de condução é necessário realizar uma inscrição. Para obter aprovação à categoria automóvel de uma inscrição, cada aluno deve realizar um exame (se reprovar no exame, terá de fazer nova inscrição). Cada exame é preparado pela Direção Geral de Viação para vários alunos: cada aluno obtém a categoria respetiva assim que o resultado do exame for definido, sendo a data do exame a que define a data de obtenção da categoria correspondente. Quando uma inscrição for paga, a data de pagamento será registada e o valor do atributo *paga* será automaticamente atualizado.

Considere que a base de dados utilizada para armazenar toda a informação contém as tabelas da Figura 1, onde as **chaves primárias estão a negrito e sublinhadas** e as ***chaves estrangeiras estão a negrito e itálico***:

ALUNO

bi	nome	morada	data_nasc	última_categoria_obtida	data_última_categoria_obtida	total_reprovacoes
10700007	Carlos Sousa	Rua das Tijoleiras	26-02-1997	*NULL*	*NULL*	2
10800008	Susana Costa	Rua da Beleza	29-08-1984	*NULL*	*NULL*	1
10900009	Filipe da Silva	Av. Vidal Pinheiro	01-01-1995	C	05-02-2019	0

INSCRICAO

cod_inscricao	data_insc	paga	data_pagamento	categoria	***bi_aluno***	***id_exame***	resultado_exame
7089	08-11-2019	S	08-12-2019	C	10900009	10901	A
7090	01-12-2019	S	04-01-2020	A	10700007	10900	R
7091	08-12-2019	S	04-01-2020	C	10800008	10901	R
7092	10-01-2020	N	*NULL*	B	11100000	*NULL*	*NULL*
7093	11-01-2020	S	12-01-2020	A	10700007	10902	R
7094	10-01-2020	S	20-01-2020	A	10700007	10903	NULL
7095	08-02-2020	N	*NULL*	D	10700007	*NULL*	*NULL*

EXAME

id	local	data	categoria
10809	Estádio da cidade	03-12-2009	C
10900	Estádio da cidade	04-12-2019	A
10901	Escola	01-01-2020	C
10902	Estádio da cidade	07-02-2020	A
10903	Escola	09-02-2020	A
10904	Centro de Testes Automóveis	12-03-2020	B
10905	Centro de Testes Automóveis	13-03-2020	A
10906	Centro de Testes Automóveis	01-04-2020	B
10907	Estádio da cidade	10-04-2020	B

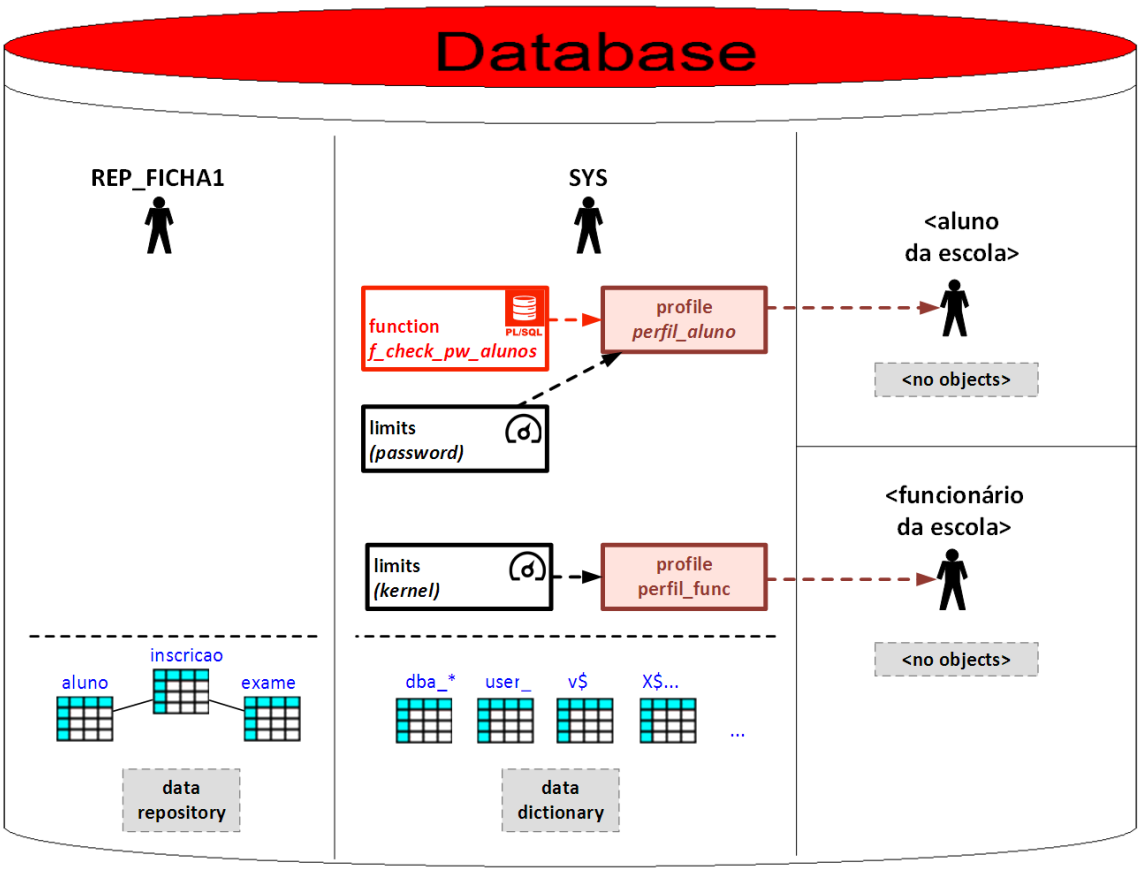

Figura 2 – Arquitetura **pretendida** ao nível da segurança.

Notas prévias

a) A presente parte da ficha 1 pressupõe que foram realizadas as tarefas da parte 1.

b) **Verifique e teste** todas as alterações realizadas sobre os objetos ou privilégios da base de dados, consultando os objetos envolvidos e/ou o dicionário de dados.

c) **Guarde num ficheiro** *sql* a sequência exata e completa dos comandos executados, juntamente com os apontamentos relevantes sobre o contexto da execução. Este ficheiro, quando executado de forma integral, deverá permitir resolver na íntegra toda a ficha.
Após terminar os exercícios, renomeie o ficheiro para <n.º_estudante_SBDficha1 parte2.sql) e submeta-o no Moodle utilizando o link apropriado (por exemplo, o estudante n.º 21000001 submeterá o ficheiro *2100001_SBDficha1_parte2.sql*).

Os exercícios assinalados com **(*)** deverão ser realizados em estudo autónomo.

Perfis de utilizador

1. Após uma auditoria de rotina à base de dados relativamente à forma como os alunos da escola de condução acedem à base de dados, o DBA apercebeu-se que os alunos da escola de condução tendem a não respeitar boas práticas na gestão das suas *passwords*. Por exemplo, 45% das alterações de *password* utilizam a *password* anterior e 82% dos alunos não altera a sua *password* durante o período em que é aluno.
Tendo em conta estas estatísticas preocupantes, o DBA irá reimplementar a política de segurança para *passwords* dos alunos da forma que é descrita de seguida:

a) A *password* de cada aluno deverá:

- Ter pelo menos 6 caracteres;
- Ser diferente da anteriormente utilizada;
- Ser diferente de "*password*" (maiúsculas ou minúsculas) e de "123456".

  (*) Defina uma forma mais dinâmica para ajudar o DBA a gerir as *passwords* que não podem ser usadas pelos alunos.

b) Para cada aluno é necessário garantir que:

- Durante o processo de *login*, o utilizador pode errar a *password* até 3 vezes consecutivas, após as quais será bloqueado durante 2 minutos;
- A *password* expira a cada 15 dias, mas o utilizador será notificado nos 5 dias que antecedem o limite de alteração;
- A *password* pode ser reutilizada, mas só após 30 dias e apenas se já tiver sido alterada pelo menos duas vezes.

c) **(*)** Descubra como poderá desbloquear-se um utilizador que tenha sido bloqueado após 3 tentativas falhadas de *login* e antes do desbloqueio automático que ocorre passados 2 minutos.

2. Relativamente aos funcionários da escola, o DBA descobriu também situações de utilização imprópria da base de dados, desta vez ao nível dos recursos do SGBD: por exemplo, 34% dos funcionários deixam as suas sessões abertas por longos períodos de tempo (+ de 150 minutos); 75% dos funcionários utiliza mais de 4 sessões em simultâneo. Numa tentativa de racionar estes e outros recursos, o DBA definiu as seguintes regras para os funcionários da escola de condução:

- Cada utilizador só pode ter 2 sessões ativas em simultâneo;
- Não há limite para o consumo de CPU em cada sessão;
- Cada comando SQL executado pode consumir até 30 segundos de CPU;
- Cada sessão pode durar até ao máximo de 2 minutos;
- No processamento de um comando SQL não pode haver uma leitura de mais de 1000 blocos de dados;
- Cada sessão não pode alocar mais de 15KB de memória na SGA;

Auditoria

3. Nas semanas anteriores à atual alguns alunos da escola reportaram alterações aparentemente não solicitadas aos seus dados. Sendo verdade, constituirá uma falha grave ao nível da segurança, pelo que é importante monitorizar a ocorrência de situações para preveni-las no futuro. Desta forma, o DBA decidiu implementar um mecanismo de auditoria específico para essa situação, registando:

- Todas as atualizações bem sucedidas a dados de alunos (1 registo de auditoria por cada atualização);
- Informação se alguma tentativa mal sucedida ocorrer (1 registo de auditoria por cada sessão que faça uma tentativa inválida).

Nota: nos testes que realizar recorde que as atualizações de dados podem ser realizadas por funcionários e pelo utilizador que detém o repositório de dados.

a) Ative, verifique e teste a auditoria da base de dados de acordo com os requisitos acima indicados.

b) **(*)** Garanta que para cada tentativa de atualização sem sucesso é registada uma entrada de auditoria (atualmente é guardado 1 registo por sessão e não 1 registo por tentativa).

c) **(*)** Ative, verifique e teste a auditoria da base de dados nas situações em que no repositório de dados sejam criados novos objetos.

d) Elimine as entradas que foram realizadas no sistema de auditoria devido às alíneas anteriores, não sem antes fazer dessas entradas para um ficheiro de texto.

e) Desligue os níveis de auditoria que definiu nas alíneas anteriores para evitar sobrecarga desnecessária do *tablespace* onde são armazenados os registos de auditoria.

Remoção de privilégios

4. **(*)** Retire aos *roles role_aluno* e *role_func* os privilégios concedidos até ao momento. Verifique e teste as alterações.

Encriptação

5. **(*)** Altere a tabela INSCRICAO, adicionando as colunas *meioPagamento* (dinheiro, cheque ou cartão) e *numCartaoCredito* (20 dígitos).

6. **(*)** Atualize as novas colunas da tabela INSCRICAO de forma coerente.

7. **(*)** Execute o script FICHA01_FUNCS.SQL de forma a criar na BD duas funções, uma para cifrar e outra para decifrar dados.

8. **(*)** Crie um bloco de código em PL/SQL que permita testar as duas funções criadas.

9. **(*)** Altere as funções FUNC_CIFRAR e FUNC_DECIFRAR de modo que a chave utilizada na cifragem seja definida no momento da chamada da função.

10. **(*)** Crie um *trigger* que cifre o atributo *numCartaoCredito* quando este é inserido ou alterado. Teste-o, inserindo uma inscrição para o aluno com o utilizador FILIPE10900009.

11. **(*)** Crie uma vista que permita consultar os exames e o número do cartão de crédito usado para pagamento das inscrições realizadas pelos alunos. Conceda privilégios de consulta desta vista ao *role* ROLE_ALUNO, de modo a que os alunos ao acederem a esta vista só possam visualizar os seus próprios dados.

12. **(*)** Ligue-se com o utilizador FILIPE10900009 e teste os privilégios concedidos ao mesmo.