POLITÉCNICO DE LEIRIA | ESCOLA SUPERIOR DE TECNOLOGIA E GESTÃO

SISTEMAS DE BASES DE DADOS

Engenharia Informática, regimes Diurno e Pós-Laboral

2.º Ano 2.º Semestre 2020/2021

FICHA 1

Segurança em Bases de Dados (parte 3)

Objetivos:

- Gerir e 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:

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

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

2. Apresente um exemplo do comando que permite 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 ser-se aluno e ter acesso às aulas de condução é necessário realizar uma inscrição. Quando uma inscrição for paga, a data de pagamento será registada e o valor do atributo *paga* será automaticamente atualizado.

Para obter aprovação à categoria automóvel de uma inscrição, cada aluno deve realizar exame: se reprovar nesse 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 aquela que define a data de obtenção da categoria correspondente.

ALUNO

bi	nome	morada	data_nasc	ult_categ_obtida	data_ult_categ_obtida	total_reprovacoes	username
1777	Carlos Sousa	Rua das Tijoleiras	26/02/1997	*NULL*	*NULL*	2	C1777
1888	Susana Costa	Rua da Beleza	29/08/1984	*NULL*	*NULL*	1	S1888
1999	Filipe da Silva	Av. Vidal Pinheiro	01/01/1995	C	05/02/2019	0	F1999

EXAME

id	local	data	categoria
10809	Estádio da cidade	03/12/2020	C
10900	Estádio da cidade	04/12/2020	A
10901	Escola	01/01/2021	C
10902	Estádio da cidade	07/02/2021	A
10903	Escola	09/02/2021	A
10904	Centro de Testes Automóveis	12/03/2020	B
10905	Centro de Testes Automóveis	13/03/2021	A
10906	Centro de Testes Automóveis	01/04/2021	B
10907	Estádio da cidade	10/04/2021	B

INSCRICAO

id	data_insc	paga	data_pagamento	categoria	***bi_aluno***	***id_exame***	resultado_exame
7089	08/11/2020	S	08/12/2020	C	1999	10901	A
7090	01/12/2020	S	04/01/2021	A	1777	10900	R
7091	08/12/2020	S	04/01/2021	C	1888	10901	R
7092	10/01/2021	N	*NULL*	B	1999	*NULL*	*NULL*
7093	11/01/2021	S	12/01/2021	A	1777	10902	R
7094	10/01/2021	S	20/01/2021	A	1777	10903	NULL
7095	08/02/2021	N	*NULL*	D	1777	*NULL*	*NULL*

Figura 1 – Modelo Lógico da Base de Dados.

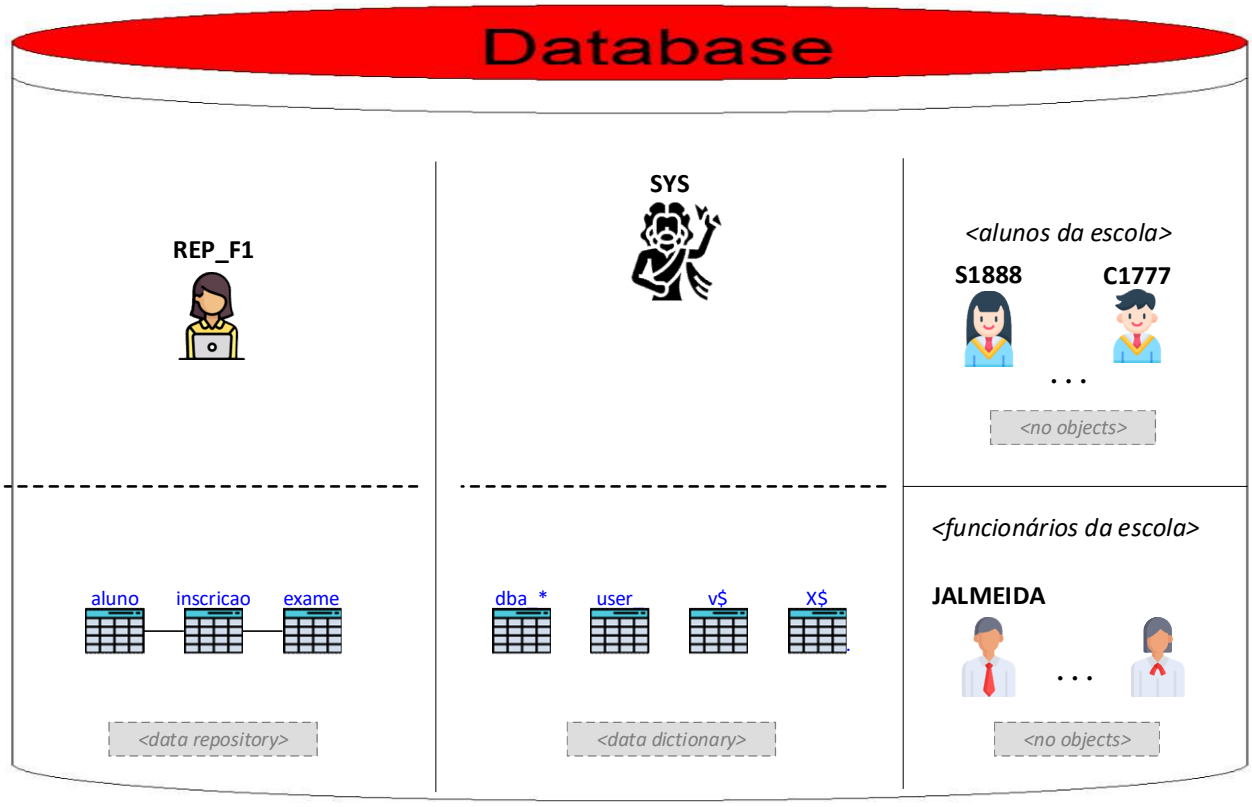

Figura 2 – Arquitetura (ao nível dos utilizadores e seus objetos).

Notas prévias

a) A presente parte da ficha 1 pressupõe que foram realizadas as tarefas das partes 1 e 2.

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>_SBDficha1.3.sql) e submeta-o no Moodle utilizando o link apropriado (por exemplo, o estudante n.º 2100001 submeterá o ficheiro *2100001_SBDficha1.3.sql*).

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

Auditoria

1. 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 desse tipo de situações para preveni-las no futuro.
Desta forma, o DBA decidiu configurar o mecanismo de auditoria da base de dados para o problema reportado. Assim, por cada atualização ou tentativa de atualização a dados de alunos, no dicionário de dados será armazenado 1 registo de auditoria.

a) Ative a auditoria da base de dados de acordo com os critérios acima indicados.

b) Verifique, utilizando o dicionário de dados, a ativação correta da auditoria.

c) Crie situações de teste que mostrem que a auditoria funciona, como por exemplo:

- Realize tentativas de atualização feitas por um *user* sem privilégios para tal;
- Realize atualizações bem-sucedidas, mas realizadas pelo utilizador dono do repositório.

2. As tentativas de acesso indevido a contas da base de dados é um problema que pode ocorrer por diversas causas, tanto devido a esquecimentos de *password* como por malícia. Uma elevada quantidade de tentativas de acesso falhadas à conta de determinado *user* pode indiciar que se está perante um ataque direcionado com o objetivo de descobrir a palavra passe desse *user* (por exemplo, por *password cracking*).

a) Dado que a conta REP_F1 contém dados e privilégios sensíveis, ative a auditoria da base de dados para identificar situações de falha nos acessos a esse *user*;

b) Simule uma situação de ataque por password *cracking* à conta REP_F1;

c) Com base nos conhecimentos de segurança já adquiridos, configure uma forma de reduzir drasticamente a eficácia destes ataques.

3. Elimine as entradas que foram realizadas no sistema de auditoria devido às alíneas anteriores, não sem antes descarregar essas entradas para um ficheiro *csv*.

4. **(*)** Ative, verifique e teste a auditoria da base de dados nas situações em que no repositório de dados sejam criadas novas tabelas.

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

Encriptação

6. **(*)** Altere a tabela INSCRICAO, adicionando as colunas *meio_pagamento* (dinheiro, cheque ou cartão) e *num_cartao_credito* (20 dígitos).

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

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

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

10. **(*)** Altere as funções F_CIFRAR e F_DECIFRAR de modo que a chave utilizada na cifragem seja definida no momento da chamada da função.

11. **(*)** Crie um *trigger* que cifre o atributo *num_cartao_credito* quando este é inserido ou alterado. Teste-o, inserindo uma inscrição para o aluno com o utilizador F1999.

12. **(*)** Crie uma vista que permita a cada aluno consultar os exames e o número (original) do cartão de crédito usado para pagamento das suas inscrições.

13. **(*)** Ligue-se com o utilizador F1999 e consulte a informação que a vista da pergunta anterior lhe disponibiliza.