Laboratoire 5, sécurité des données

Objectifs :

1. Introduction à la sécurité des données
    a. Notion de rôles
    b. Notion de privilèges et permissions
    c. Hachage des mots de passe.

Consignes :

1. Donnez-vous le rôle sysadmin pour pouvoir créer les connexions. Pour vous donner le rôle sysadmin, loguer vous avec une connexion « Authentification Windows »
2. Vous devez écrire les requêtes SQL dans un fichier Laboratoire5.sql. Lorsqu'il s'agit de répondre à une question, vous devez répondre dans le même fichier en mettant la réponse en commentaires.
3. Vous devez utiliser le code SQL pour faire le laboratoire.
4. Pour répondre aux questions, exécutez des requêtes appropriées pour vérifier votre réponse.

Exercice no1 :

1. Créer une nouvelle connexion de nom **vosInitialDarQuest**.(**exemple : SYDarQuest1**) Cette connexion a le role public uniquement.
    a. Cette connexion a un utilisateur mappé de même nom et utilise la base de données de votre Tp 1.(DarQuest).
    b. L'utilisateur mappé a les roles : db_datareader et db_datawriter.(sur la BD)
2. En écrivant les requêtes, vérifier que votre utilisateur a les privilèges suivants sur les toutes les tables de votre base de données :
    a- Privilège SELECT
    b- Privilège INSERT, UPDATE et DELETE.
3. Votre utilisateur a-t-il le droit de créer des tables dans votre base de données ? si vous répondez non, donnez-lui le role ou autorisations nécessaires pour qu'il puisse le faire.
4. Votre utilisateur a-t-il le droit d'exécuter des procédures stockées ? si vous répondez non, donnez-lui le droit de le faire.

Exercice no2

1. Créer une nouvelle connexion de nom **vosInitialDarQuest2**. Cette connexion a le role public uniquement.
    a. Cette connexion a un utilisateur mappé de même nom et utilise la base de données de votre Tp 1.(Darquest).
    b. L'utilisateur mappé a le role public uniquement.
2. Concrètement que peut faire votre connexion **vosInitialDarQuest2** sur la base de données ?

3. Pour votre utilisateur **vosInitialDarQuest2**, donnez les privilèges suivants :
    a- SELECT sur toutes les tables.
    b- INSERT sur les tables Items, Armes,Potions, Element et Armures
    c- UPDATE de qtStock, flagDispo de la table *Items*
    d- D'exécuter les procédures : ajouterArmes, ajouterPotions, ajouterArmure,ajouterElement

4. Pour cette question, créer une connexion de nom **loginAlias.** Où Alias désigne l'Alias d'un de vos joueurs. Exemple **loginPatoche.**
    a. Cette connexion a un utilisateur mappé de même nom et utilise la base de données de votre Tp 1.
    b. L'utilisateur mappé a le role public uniquement.
    c. Créer une vue VJoueurs sur la table Joueurs pour un joueurs dont l'Alias est 'Patoche' (prenez un alias de votre choix) . Cette vue doit être créé avec l'option WITH CHECK OPTION.
    d. Donner le droit SELECT, et UPDATE à votre utilisateur sur cette vue.
    e. Vérifier que 'Patoche' (ou le joueur de votre choix) ne peux pas voir, ni mettre à jour les données des autres joueurs.

5. Créer une nouvelle connexion **vosInitialDarquest3,** mappée sur un utilisateur de même nom et qui utilise votre BD du Tp 1. La connexion, comme l'utilisateur ont le role PUBLIC.
6. Créer un role **roleDEV.** Pour ce role attribuer les droits SELECT et INSERT sur toutes vos tables. Le droit UPDATE du prix, de la quantité en stock de la table Items, le droit UPDATE *EstAlchimiste* de la table joueurs et de DELETE de la table Paniers.
7. Attribuer ce role à votre utilisateur **vosInitialDarquest3.**
8. Enlever le droit SELECT sur la table joueurs à ce user.

Exercice no3

1. Pour la table joueurs, ajouter la colonne **mPasse** varbinary (128). Cette colonne a pour but d'accueillir les mots de passes hachés de vos joueurs
2. Mettre à jour cette colonne avec des mots de passe hachés.
3. Insérer deux nouveaux joueurs de votre choix.
4. Écrire la procédure stockée qui permet à un joueur de modifier son mot passe. Cette procédure reçoit comme paramètre l'Alias du joueurs, l'ancien mot de passe et le nouveau mot de passe.
5. Écrire la procédure stockée qui permet d'insérer un joueur avec un mot de passe haché.