Séance de Projet

Sécurisation d'un SI

L'objectif de ce projet est de voir les différentes options permettant de sécuriser une base de données
Pour ce faire, vous manipulerez une base de données MySQL:

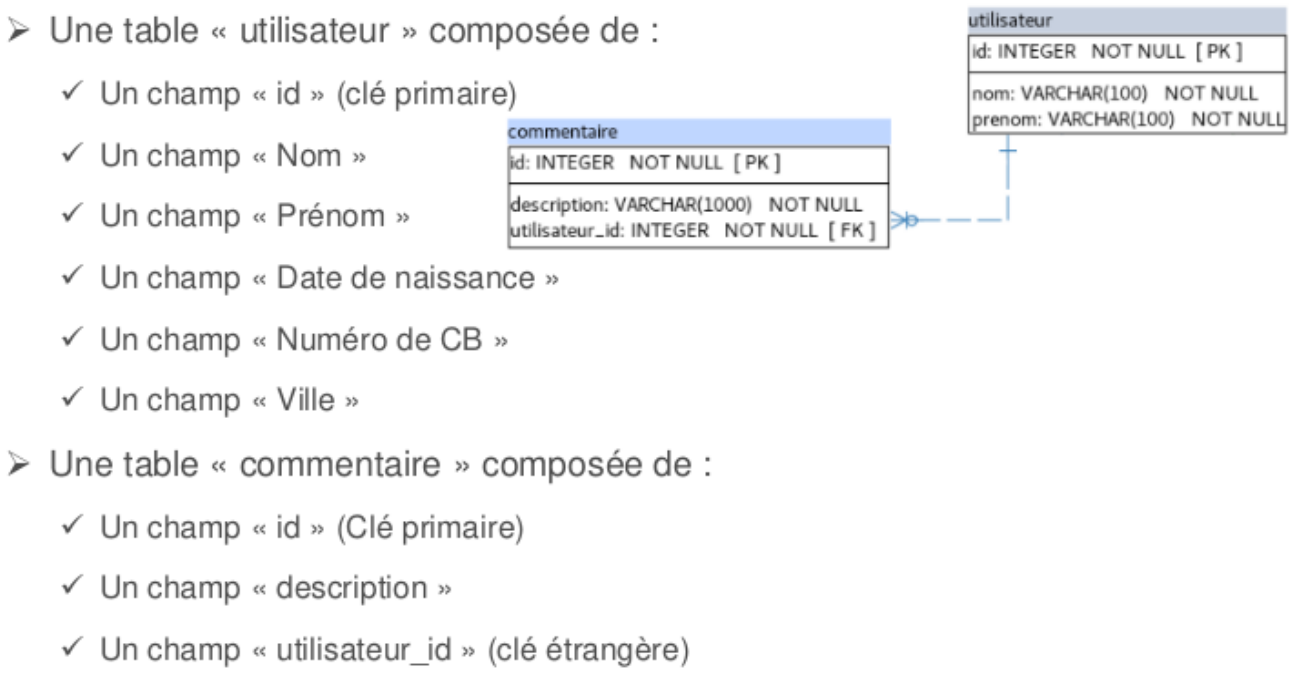
- Création d'utilisateurs et gestion des droits
- Création de base de données et de tables
- Création de relation entre les tables et tests d'accès aux données
- Création de vue et test d'accès aux données
- Activation de la journalisation
- Sauvegarde et restauration

Exercice I - Gestion des droits

1. Installer Wamp sur votre poste de travail (ou XAMPP sur Mac ou Linux) (http://www.wampserver.com/)
2. Créer une base de données MySQL
3. Créer un utilisateur pour cette base
4. Ne donner à l'utilisateur (SGBD) de l'application que les droits nécessaires
    a. SELECT, INSERT, UPDATE et DELETE sur la base de données précédemment créée
5. Créer un second utilisateur et lui accorder des droits de lecture uniquement sur la base de données

Exercice II - Relation entre les données

1. Depuis l'interface d'administration, créer deux tables :



2. Créer un script PHP accédant à la base de données
3. A partir du script PHP en utilisant PDO :
    a. Créer une première page Web contenant un formulaire permettant de créer dynamiquement des utilisateurs en évitant les injections SQL
    b. Créer trois utilisateurs dans la table « utilisateur » en utilisant le formulaire
    c. Affichez les données correspondant aux utilisateurs en empêchant toute
    d. Exécution de code potentiellement stocké dans les valeurs de la base
    e. Créer un bouton permettant de supprimer un utilisateur
    f. Créer une seconde page web affichant l'ensemble des commentaires de la table « commentaire » en empêchant toute exécution de code potentiellement stocké dans les valeurs de la base.
    g. Insérer un commentaire dans la table « commentaire » avec un identifiant d'utilisateur correspondant à un utilisateur.
    h. Insérer un commentaire dans la table « commentaire » avec un identifiant d'utilisateur qui n'existe pas dans la table « utilisateur ». Comment réagit la base de données?
    i. Supprimer l'utilisateur dont l'identifiant a été utilisé dans la table commentaire. Comment réagit la base de données?
4. Afin d'empêcher le second utilisateur d'accéder à des données sensibles (numéro de CB), créer une vue de la table utilisateur
    a. La vue de la table utilisateur ne contiendra que l'identifiant, le nom, le prénom et la ville
5. Retirer le droit de lecture de la table utilisateur au second utilisateur
6. Donner accès à la vue au second utilisateur
    a. Depuis le script PHP, en tant que second utilisateur, afficher les données de la vue

Exercice III - Audit et Sauvegarde

1. Activer l'audit des requêtes
    a. Accéder aux logs précédemment activées : retrouver les requêtes
2. Créer une sauvegarde de la base de données
    a. Réaliser des modifications quelconques dans la base
3. Restaurer la sauvegarde
    a. Vérifier que les modifications ont disparues