# SÉCURITÉ DES BASES DE DONNÉES DANS LE CONTEXTE DES applications web.

## ACTIVITÉ DE RECHERCHE

ANNÉE: 2023/2024

# **PARTIE 01 :** MENACE ET VULNÉRABILITÉ

## Définition d'une menace :

Une menace en base de données dans le contexte des applications web concernées tout ce qui pourrait mettre en danger l'intégrité, la confidentialité, ou la disponibilité des données stockées dans la base de données utilisée par l'application.

## Définition d'une vulnérabilité :

Une vulnérabilité fait référence à une faiblesse, une lacune ou une faille de sécurité potentielle dans le système ou le code source qui peut être exploitée par des attaquants pour nuire à la sécurité de l'application et de la base de données associées. Les vulnérabilités peuvent résulter d'erreurs de programmation, de configurations incorrectes, de défauts de conception ou de l'utilisation de versions obsolètes de logiciels.

## Différence entre menace et vulnérabilité :

Une menace est un danger potentiel qui pourrait nuire à la sécurité des données stockées dans une base de données, tandis qu'une vulnérabilité est une faiblesse ou une faille dans le système ou le code source de l'application qui pourrait être exploitée par une menace pour nuire à la sécurité. En d'autres termes, la menace est le danger, et la vulnérabilité est la porte d'entrée par laquelle ce danger peut se concrétiser. La gestion efficace des vulnérabilités est essentielle pour atténuer les menaces et renforcer la sécurité globale d'un système.

| menaces | description | exemple |
|---|---|---|
| Injection SQL | Consiste à modifier une requête SQL en cours par l'injection d'un morceau de requête non prévu, souvent par le biais d'un formulaire. Le hacker peut ainsi accéder à la base de données, mais aussi modifier le contenu et donc compromettre la sécurité du système. | Supposons qu'un site web utilise une requête SQL pour vérifier l'authentification d'un utilisateur lors de la connexion. La requête SQL: SELECT * FROM utilisateurs WHERE nom_utilisateur = 'nom' AND mot_de_passe = 'motdepasse';<br><br>Après l'injection, la requête modifiée ressemblera à ceci :SELECT * FROM utilisateurs WHERE nom_utilisateur = 'nom' OR '1'='1'; --' AND mot_de_passe = 'motdepasse';<br>Dans cette situation, la condition '1'='1'est toujours vraie, ce qui signifie que la requête renverra toutes les lignes de la table des utilisateurs, permettant à l'attaquant de contourner l'authentification et d'accéder à un compte sans connaître le mot de passe. |

| | | |
|---|---|---|
| Accès non autorisé | Des tentatives d'accès non autorisées peuvent être menées par des attaquants cherchant à exploiter des failles d'authentification ou de contrôle d'accès pour obtenir des informations sensibles | |
| Attaques par force brute | Des attaquants peuvent tenter de deviner des identifiants en imposant différentes combinaisons de noms d'utilisateur et de mots de passe jusqu'à ce qu'ils trouvent les bonnes. | |
| Attaques de contournement d'authentification | Des attaquants peuvent exploiter des faiblesses dans les mécanismes d'authentification pour contourner les mesures de sécurité et accéder à des données sensibles. | |
| Déni de service Dos | Visent à rendre la base de données indisponible en surchargeant les ressources du serveur avec un grand nombre de requêtes. | -L'attaquant envoie un grand nombre de paquets SYN au serveur cible.<br>-Le serveur crée des connexions en attente pour chaque paquet SYN reçu.<br>-L'attaquant ne répond pas par des paquets ACK.<br>-Le serveur finit par manquer de ressources pour gérer de nouvelles connexions, rendant le service inaccessible pour les utilisateurs légitimes. |
| Mauvaise gestion des droits d'accès | Lorsque les autorisations pour accéder à des informations ne sont pas correctement définies, laissant soit trop de personnes y accéder, soit ne pas à ceux qui en ont besoin d'y accéder. | |
| | Cela peut causer des problèmes de sécurité. | |
| Fuites d'informations sensibles | Quand des informations secrètes ou privées sont accidentellement révélées ou volées. Cela peut arriver si les protections informatiques ne sont pas assez bonnes, si quelqu'un fait une erreur, ou si quelqu'un essaie régulièrement de voler des informations importantes. | |

| | | |
|---|---|---|
| Falsification de requête(CRSF Cross-Site Request Forgery) | Un attaquant manipule les données d'une requête HTTP afin de tromper l'application web et d'effectuer des actions non autorisées. Cette technique d'attaque peut être utilisée pour modifier des paramètres, accéder à des informations sensibles, ou effectuer des opérations indésirables. | Supposons qu'un site web utilise une requête GET pour effectuer une action de changement de mot de passe, et la requête ressemble à ceci : https://exemple.com/changer_mot_de_passe?utilisateur=123&nouveau_mot_de_passe=nouveau123<br><br>l'identifiant est "123", et le nouveau mdp est "nouveau123". Par exemple, l'attaquant pourrait modifier l'identifiant de l'utilisateur pour cibler un compte différent : https://exemple.com/changer_mot_de_passe?utilisateur=456&nouveau_mot_de_passe=nouveau456<br>Si l'application web ne met pas en place des mécanismes de protection appropriés, elle pourrait effectuer le changement de mdp pour l'utilisateur avec l'identifiant « 456 », même si l'attaquant n'a pas l'autorisation d 'effectuer cette action. |

| Vulnérabilités | Description |
|---|---|
| Gestion des erreurs inefficace | Des messages d'erreur détaillés peuvent fournir des informations sensibles aux attaquants, facilitant ainsi les attaques. |
| Faible cryptage des données | Une mauvaise gestion de la confidentialité des données, y compris un cryptage inadéquat, peut exposer des informations sensibles en cas de violation de sécurité. |
| Mauvaise gestion des sessions | Les vulnérabilités liées à la gestion des sessions, telles que la fixation de session, peuvent nuire à l'authentification des utilisateurs. |
| Logiciels obsolètes | L'utilisation de versions obsolètes de logiciels de base de données ou de serveurs web peut laisser des failles de sécurité non corrigées, exposant la base de données à des risques. |
| Manque de contrôles d'accès | Une gestion insuffisante des autorisations et des contrôles d'accès peut permettre à des utilisateurs non autorisés d'accéder à des parties sensibles de la base de |
| | données. |

# **PARTIE 02 :** Stratégies de sécurité

# I. Pratiques pour sécuriser une base de données

1. **Authentification forte :** Mettez en place une authentification forte, nécessitant au moins deux facteurs pour accéder à une base de données. L'authentification forte repose sur l'utilisation de deux facteurs ou plus pour vérifier l'identité d'un utilisateur. Ces facteurs peuvent appartenir à trois catégories :
   - **Quelque chose que vous savez :** Généralement un mot de passe ou un code PIN.
   - **Quelque chose que vous avez :** Par exemple, un appareil mobile, une carte à puce, ou un jeton d'authentification.
   - **Quelque chose que vous êtes :** Les caractéristiques biométriques telles que les empreintes digitales, la reconnaissance faciale, ou l'iris.

   Pour mettre en œuvre une authentification forte on peut utiliser :
   - **Applications d'Authentification :** L'utilisation d'applications mobiles générant des codes temporaires.
   - **Cartes à Puce :** Les cartes à puce contenant des informations d'authentification.
   - **Biométrie :** Intégration de caractéristiques biométriques pour vérification.

   Il est important aussi que les codes ou les facteurs d'authentification peuvent avoir une durée de validité limitée, renforçant la sécurité en réduisant le risque d'utilisation frauduleuse.

2. **Gestion des mots de passe :** Lorsqu'un utilisateur crée un compte ou modifie son mot de passe, le mot de passe ne doit jamais être stocké en texte brut dans la base de données. Au lieu de cela, il est haché à l'aide d'algorithmes de hachage robustes tels que bcrypt, Argon2, ou scrypt. Ces algorithmes sont conçus pour être lents et coûteux, rendant difficile pour les attaquants de décrypter les mots de passe en utilisant des attaques de force brute ou des attaques par dictionnaire.

   - **Complexité des Mots de Passe :**

   Les politiques de complexité des mots de passe, telles que la longueur minimale, l'utilisation de lettres majuscules, de chiffres et de caractères spéciaux, devraient être mises en œuvre pour encourager la création de mots de passe forts.
   Cela réduit la probabilité d'attaques par force brute en augmentant l'espace des combinaisons possibles.

   - **Gestion des Mises à Jour de Mots de Passe :**

   Les utilisateurs devraient être encouragés à mettre à jour régulièrement leurs mots de passe.

3. **Contrôle d'accès :**

   Le contrôle d'accès est le processus de régulation de l'entrée ou de l'utilisation de ressources dans un système. Il englobe plusieurs composantes :

   - **Identification :**

   Les utilisateurs doivent être identifiables de manière unique dans le système. Cela est généralement réalisé à l'aide d'identifiants, tels que des noms d'utilisateurs.

- **Authentification :**

Le processus par lequel le système vérifie l'identité déclarée de l'utilisateur. Cela implique souvent l'utilisation de mots de passe, mais peut également inclure des méthodes telles que l'authentification à deux facteurs (2FA) ou l'utilisation de certificats.

- **Autorisation :**

Une fois authentifié, l'utilisateur doit être autorisé à accéder à certaines ressources ou fonctionnalités. Cela implique la définition des droits et des permissions associés à chaque utilisateur ou groupe d'utilisateurs.

- **Contrôle en Fonction du Contexte :**

Les systèmes modernes intègrent souvent des mécanismes de contrôle d'accès en fonction du contexte, prenant en compte des facteurs tels que l'emplacement de l'utilisateur, le type d'appareil utilisé, etc.

- **Révision des Accès :**

Les droits d'accès doivent être régulièrement révisés pour s'assurer qu'ils correspondent toujours aux besoins opérationnels et pour révoquer l'accès lorsque cela n'est plus nécessaire.

4. **Autorisation :**

L'autorisation est le processus de détermination des actions spécifiques qu'un utilisateur autorisé est autorisé à effectuer sur une ressource ou dans un système. Cela implique des décisions basées sur les droits et les permissions associés à l'utilisateur. Quelques concepts clés liés à l'autorisation incluent :

- **Droits d'Accès :**

Les droits d'accès définissent les actions spécifiques qu'un utilisateur est autorisé à effectuer. Par exemple, un utilisateur peut avoir le droit de lecture, d'écriture, de suppression, etc.

- **Permissions :** Les permissions détaillent quelles ressources spécifiques un utilisateur peut accéder. Cela peut inclure des fichiers, des dossiers, des fonctionnalités spécifiques d'une application, etc.
- **Groupes d'Utilisateurs :**

Pour simplifier la gestion des autorisations, les utilisateurs sont souvent regroupés en fonction de leurs rôles ou responsabilités. Les autorisations sont ensuite définies pour ces groupes plutôt que pour chaque utilisateur individuellement.

5. **Chiffrement de données :**

Le chiffrement des données est une pratique essentielle en matière de sécurité informatique qui vise à protéger les informations sensibles en les rendant illisibles pour toute personne non autorisée. Il est utilisé dans divers contextes, y compris le stockage de données, la transmission de données sur un réseau et d'autres scénarios où la confidentialité des informations est cruciale.

-Chiffrer les données sensibles, en particulier lorsqu'elles sont stockées sur des disques ou transitent sur des réseaux non sécurisés.

- Utilisez le chiffrement au repos et en transit pour garantir la confidentialité des informations stockées et échangées :

- **Chiffrement de Données en Repos (Data-at-Rest) :**

Les données stockées sur des disques durs, des serveurs ou d'autres supports de stockage sont chiffrées pour empêcher un accès non autorisé en cas de vol physique ou d'accès non autorisé.

- **Chiffrement de Données en Transit (Data-in-Transit) :**

Les données qui sont transférées sur un réseau, comme lors de connexions HTTPS, sont chiffrées pour protéger contre l'interception de données par des tiers non autorisés.

6. **Surveillance (Monitoring) :**

La surveillance consiste à observer de manière continue les activités, les performances et les comportements d'un système ou d'un réseau pour identifier les incidents de sécurité potentiels, les anomalies, ou les tendances significatives.

- **Sources de Surveillance :**

La surveillance peut se baser sur diverses sources, telles que les journaux d'événements (logs), les alertes de sécurité, les données de trafic réseau, les comportements des utilisateurs, les performances du système, etc.

- **Surveillance en Temps Réel :**

La surveillance en temps réel permet la détection immédiate des incidents ou des comportements anormaux, permettant une réponse rapide aux menaces.

- **Surveillance des Utilisateurs :**

La surveillance des utilisateurs peut inclure la détection de tentatives d'accès non autorisées, de modifications inattendues des autorisations, ou de comportements atypiques.

- **Surveillance des Applications :**

Les applications peuvent être surveillées pour détecter des activités anormales, des tentatives d'intrusion, ou des vulnérabilités potentielles.

- **Surveillance du Trafic Réseau :**

La surveillance du trafic réseau permet de détecter les attaques par déni de service (DoS), les scans de ports, les tentatives d'intrusion, ou tout comportement inhabituel.

7. **Mise à jour régulières :**

Les mises à jour régulières, également appelées patching, sont un aspect fondamental de la gestion de la sécurité des systèmes informatiques. Elles consistent à appliquer les correctifs de sécurité, les mises à jour logicielles et les mises à jour du système d'exploitation de manière régulière pour garantir la stabilité, la performance et surtout la sécurité du système.

8. **Sauvegarde régulière :**
   Les sauvegardes régulières constituent une pratique cruciale en matière de gestion de la sécurité des données et de préparation aux incidents. Elles consistent à copier et à stocker des copies de données importantes à des intervalles réguliers dans le but de les restaurer en cas de perte de données, de corruption ou d'autres incidents.

9. **Formation et sensibilisation :**
   La formation et la sensibilisation en matière de sécurité informatique sont des composantes cruciales pour renforcer la posture de sécurité d'une organisation. Ces initiatives visent à éduquer les employés sur les meilleures pratiques en matière de sécurité, à les sensibiliser aux menaces potentielles et à les habiliter à contribuer activement à la protection des informations de l'entreprise.

# II. Comment éviter les attaques d'injection SQL

1. **Utilisation de Requêtes Paramétrées :**
   Préférer l'utilisation de requêtes paramétrées au lieu de concaténer des chaînes de caractères pour former des requêtes SQL. Les requêtes paramétrées utilisent des paramètres définis et sécurisés pour éviter les injections.

2. **Validation des Entrées Utilisateurs :**
   Valider et filtrer rigoureusement toutes les entrées utilisateurs. Limiter les caractères autorisés et utiliser des expressions régulières pour valider les formats attendus.

3. **Principe du Moindre Privilège :**
   Accorder aux utilisateurs et aux applications uniquement les permissions nécessaires pour effectuer leurs tâches. Éviter d'accorder des privilèges excessifs.

4. **Éviter la Concaténation de Chaînes :**
   Éviter de concaténer des chaînes pour créer des requêtes SQL. Utiliser plutôt des procédures stockées ou des requêtes paramétrées.

5. **Échappement des Caractères Spéciaux :**
   Échapper aux caractères spéciaux dans les entrées utilisateur. Cela signifie traiter les caractères potentiellement dangereux de manière à les rendre inoffensifs.
6. **Utilisation de Pare-feu d'Application Web (WAF) :**
   Mettre en place un pare-feu d'application web pour filtrer et bloquer les requêtes malveillantes avant qu'elles n'atteignent la base de données.

7. **Surveillance des Logs :**
   Surveiller régulièrement les logs du système et de la base de données pour détecter toute activité suspecte ou tentative d'injection SQL.

# **Partie 03 :** Protection des données sensibles

**5-** Le chiffrement et le masquage des données sensibles stockées dans les bases de données sont des pratiques cruciales pour renforcer la sécurité des informations. Voici une analyse des méthodes de chiffrement et de masquage des données sensibles :

## Le chiffrement de données :

1. **Chiffrement Symétrique :** Utilisation d'une seule clé symétrique partagée pour le chiffrement et le déchiffrement des données, offrant une sécurité efficace mais nécessitant une gestion rigoureuse des clés.

2. **Chiffrement Asymétrique :** Utilisation de deux clés distinctes (publique et privée) pour le chiffrement et le déchiffrement, offrant une solution sécurisée pour la transmission de données, en particulier pour de petites quantités d'informations.

## Méthodes de Chiffrement :

| | |
|---|---|
| AES (Advanced Encryption Standard) | Algorithme symétrique qui chiffre des blocs de données à l'aide de clés de 128, 192 ou 256 bits, largement utilisé pour sécuriser le Wi-Fi, les applications mobiles, les dossiers et les connexions Internet. |
| RSA (Rivest-Shamir-Adleman) : | Algorithme asymétrique basé sur la factorisation de grands nombres premiers, efficace pour sécuriser la transmission de données, mais peut présenter des limites pour le chiffrement de grands volumes. |
| Triple DES (Data Encryption Standard) | Technique symétrique avancée, applique trois fois l'algorithme DES sur chaque bloc de données, utilisé pour chiffrer des codes aux guichets automatiques, mots de passe UNIX, et applications comme Microsoft Office. |
| Blowfish | Algorithme symétrique rapide, flexible et robuste, conçu pour remplacer DES, souvent utilisé pour sécuriser les |

| | |
|---|---|
| | transactions sur les plateformes de commerce électronique et les gestionnaires de mots de passe. |
| Twofish | Technique symétrique sans licence, successeur polyvalent de Blowfish, chiffre des blocs de données de 128 bits en 16 tours, adapté à diverses applications de chiffrement. |
| FPE (Format-Preserving Encryption) | Algorithme symétrique préservant le format et la longueur des données pendant le chiffrement, utile pour sécuriser les logiciels et les outils de gestion du cloud. |
| ECC (Elliptic Curve Cryptography) | Cryptographie à clé publique plus efficace que RSA, utilisant des clés plus courtes, adaptée aux protocoles tels que SSL/TLS, signatures électroniques, et le cryptage d'e-mails. |

## Masquage des données :

Le masquage des données consiste à masquer les données en modifiant leurs lettres et leurs chiffres d'origine de manière à les rendre partiellement ou complètement anonymes. Cette technique aide les entreprises à respecter les réglementations relatives à la confidentialité des données telles que le règlement général sur la protection des données (RGPD). Elles permettent de protéger de nombreux types de données, comme les données d'identification personnelle (PII), les données financières, les informations protégées sur la santé (PHI) et la propriété intellectuelle.

### Les différents type de masquage de données:

- **Masquage dynamique :** Le masquage dynamique permet de masquer certaines parties des données en temps réel, en fonction des autorisations de l'utilisateur. Par exemple, les numéros de carte de crédit peuvent être masqués pour les utilisateurs qui n'ont pas besoin d'y accéder.

- **Masquage statique :** Le masquage statique consiste à masquer des parties spécifiques des données de manière permanente. Par exemple, masquer les derniers chiffres d'un numéro de sécurité sociale.

- **Masquage partiel :** Il s'agit de masquer certaines parties des données tout en laissant d'autres visibles. Cela peut être utile pour protéger l'information tout en préservant l'utilité de certaines données.

## Techniques de masquage de données :

- **Randomisation :** Remplacement de données sensibles par des valeurs générées aléatoirement, préservant l'anonymat en utilisant des informations fictives ou aléatoires.

- **Substitution :** Remplacement de données sensibles par des équivalents fictifs ou issus d'une liste prédéfinie, préservant la structure tout en protégeant les informations réelles.

- **Mélange :** Réorganisation aléatoire des données pour préserver les propriétés statistiques et masquer les enregistrements individuels, tout en conservant les relations au sein du jeu de données.

- **Chiffrement :** Conversion des données sensibles en un format crypté, nécessitant des clés de déchiffrement pour accéder aux informations, renforçant la sécurité au détriment des performances.

- **Hachage :** Transformation des données en une chaîne de caractères de longueur fixe, souvent utilisée pour masquer les mots de passe sans nécessiter la récupération des données d'origine.

- **Création de jeton :** Remplacement des données par un jeton généré aléatoirement, conservant l'intégrité des données tout en minimisant le risque d'exposition des informations sensibles.

- **Annulation :** Remplacement des données sensibles par des valeurs nulles ou des espaces vides, préservant la structure tout en éliminant efficacement les informations spécifiques.

**6-** La conformité au RGPD (Règlement Général sur la Protection des Données) est cruciale pour diverses raisons.

- ❖ Elle protège les droits des individus en leur accordant un contrôle accumulé sur leurs données personnelles.
- ❖ Renforce la confiance des consommateurs et responsabilise les entreprises.
- ❖ La conformité réduit le risque de sanctions financières et atténue l'impact négatif sur la réputation en cas de violation.

- Elle encourage l'innovation tout en respectant la vie privée, assure une réglementation uniforme dans l'UE, gère les risques liés à la protection des données, et promet la transparence dans le traitement des informations personnelles. En somme, elle est essentielle pour établir un équilibre entre les intérêts commerciaux et le respect de la vie privée des individus.