Partie A – Analyse approfondie

1.1 - Fonctionnement global du script PHP

Le script PHP fourni permet d'effectuer une recherche de livres dans une base de données en fonction d'un mot-clé saisi par l'utilisateur. Voici son fonctionnement détaillé :

- **Connexion à la base de données** : Le script utilise mysqli_connect() pour se connecter à la base de données MySQL en utilisant un nom d'utilisateur et un mot de passe.
- **Récupération du paramètre search** : La valeur du paramètre search est récupérée directement depuis l'URL via $_GET['search'].
- **Construction de la requête SQL** : La requête SQL est construite dynamiquement en insérant directement la valeur du paramètre search dans la requête :
    - $sql = "SELECT * FROM livres WHERE titre LIKE '%" . $search . "%'";
    - Cela signifie que toute valeur saisie par l'utilisateur sera intégrée dans la requête SQL sans vérification ni protection.
- **Exécution de la requête** : mysqli_query($conn, $sql) exécute la requête SQL et stocke le résultat.
- **Affichage des résultats** : Les résultats sont affichés directement en insérant les valeurs issues de la base de données dans du HTML.

1.2 - Définition et explication des vulnérabilités

Injection SQL

Une injection SQL est une attaque qui consiste à insérer du code SQL malveillant dans une requête afin de manipuler la base de données. Elle peut permettre à un attaquant de :

- Lire, modifier ou supprimer des données non autorisées.
- Contourner les systèmes d'authentification.
- Exécuter des commandes SQL arbitraires.

Cross-Site Scripting (XSS)

Le **XSS (Cross-Site Scripting)** est une attaque qui consiste à injecter du code JavaScript malveillant dans une page web, généralement via un champ de saisie utilisateur, afin d'exécuter des actions malicieuses comme :

- Voler des cookies et usurper une session.

- Afficher de fausses informations à l’utilisateur.
- Rediriger l’utilisateur vers un site malveillant.

Les vulnérabilités identifiés sont les suivantes :

Dans le script, le paramètre search est inséré directement dans la requête SQL sans être filtré ni échappé. Un attaquant peut injecter du SQL malveillant . Aussi les résultats sont affichés directement sans être filtrés. Si un attaquant insère un script JavaScript .

Mauvaise gestion des mots de passe et sécurité des emails

D’après les informations données, il est mentionné que la gestion des mots de passe et des emails présente des faiblesses, ce qui peut inclure :

- Des politiques de mots de passe faibles (ex : pas d’exigence de longueur ou de complexité).
- L’absence de hachage sécurisé des mots de passe (stockage en clair).
- Aucun mécanisme de vérification des emails (ex : pas de double confirmation, possibilité d’inscription avec des adresses fausses).

Ces faiblesses peuvent conduire à des compromissions de comptes et des attaques par force brute ou phishing.

1.3 - Analyse des logs et confirmation des failles

Les logs montrent des tentatives d’exploitation des vulnérabilités :

1. **Injection SQL**
    - Exemple d’entrée dans les logs
        - "' OR '1'='1"

Cela montre qu’un attaquant a tenté de manipuler la requête SQL pour récupérer toutes les données de la table livres.

2. **XSS**

- Exemple d’entrée dans les logs
    - "<script>alert('Test XSS')</script>"

cela montre qu’un attaquant a essayé d’injecter un script JavaScript pour exécuter du code malveillant dans le navigateur des utilisateurs.

Ces logs prouvent que les vulnérabilités ne sont pas seulement théoriques mais bien exploitées activement par des attaquants.

2.1 - Exigences du RGPD/CNIL dans le contexte de Biblionet

Le RGPD (Règlement Général sur la Protection des Données) et les recommandations de la CNIL (Commission Nationale de l'Informatique et des Libertés) imposent des règles strictes pour la protection des données personnelles des utilisateurs.

Dans le cas de Biblionet, qui propose un service en ligne de gestion de livres et de réservations, cela implique plusieurs obligations :

- **Collecte et traitement des données personnelles**
  - Les données personnelles ne doivent être collectées que pour un but précis et légitime (principe de finalité).
  - L'utilisateur doit être informé de la manière dont ses données seront utilisées et doit donner son consentement (transparence).
- **Sécurité et protection des données**
  - Les données personnelles doivent être protégées contre les accès non autorisés (confidentialité).
  - L'application doit mettre en place des mesures de sécurité comme le chiffrement des données sensibles et la sécurisation des accès (intégrité).
- **Droits des utilisateurs**
  - Les utilisateurs doivent pouvoir accéder à leurs données, les modifier ou les supprimer (droit d'accès, de rectification et d'effacement).
  - Ils doivent pouvoir récupérer leurs données sous un format exploitable (portabilité).
  - Ils ont le droit de s'opposer au traitement de leurs données dans certains cas.
- **Notification en cas de violation de données**
  - En cas de fuite ou de compromission des données personnelles, Biblionet doit informer la CNIL et les utilisateurs concernés dans les meilleurs délais.

2.2 - Données personnelles à protéger et mise en conformité

Données personnelles concernées dans Biblionet :

1. Identifiants des utilisateurs (nom, prénom, email, pseudonyme).
2. Données de connexion et d'authentification (mots de passe, adresse IP, logs de connexion).
3. Données de réservation et d'historique (livres consultés, réservations effectuées).
4. Données de contact (email, éventuellement numéro de téléphone).

Mesures à mettre en place pour être conforme :

Sécurisation des mots de passe

- Stocker les mots de passe sous forme de hachage sécurisé **(**bcrypt, Argon2**)**.
- Appliquer des politiques strictes (longueur minimale, caractères spéciaux).

Sécurisation des données en transit et en stockage

- Utiliser HTTPS pour sécuriser les échanges entre l'application et les utilisateurs.
- Chiffrer les données sensibles stockées en base.

Gestion des accès et authentification

- Imposer une authentification forte (ex : double authentification 2FA).
- Restreindre l'accès aux données aux seuls employés autorisés.

Consentement et transparence

- Informer clairement l'utilisateur de l'usage de ses données (politique de confidentialité).
- Demander le consentement explicite avant toute collecte de données non essentielle.

Mise en place des droits RGPD

- Fournir une interface permettant aux utilisateurs de consulter, modifier ou supprimer leurs données personnelles.
- Mettre en place un contact dédié pour les demandes RGPD.

J**ournalisation et surveillance des accès**

- Conserver un historique des connexions et des actions sensibles pour détecter toute activité suspecte.

- Mettre en place une notification en cas de fuite de données

3 . Réflexion sur la sécurité globale

Dans un contexte où les cyberattaques sont de plus en plus fréquentes, il est essentiel pour une entreprise comme Biblionet d'adopter une approche globale de la sécurité. Cela signifie qu'il ne suffit pas seulement de corriger les vulnérabilités techniques dans le code, mais aussi de mettre en place des mesures de protection à différents niveaux.

Sur le plan technique, il est crucial de sécuriser le code en utilisant des bonnes pratiques comme les requêtes préparées pour éviter l'injection SQL et l'échappement des données pour prévenir les attaques XSS. L'infrastructure doit également être renforcée avec des protocoles comme HTTPS, un bon paramétrage des serveurs et des mises à jour régulières des logiciels utilisés.

D'un point de vue organisationnel, la gestion des mots de passe doit être rigoureuse en imposant un minimum de complexité et en les stockant de manière sécurisée (hachage avec bcrypt ou Argon2). De plus, la protection des emails est essentielle pour éviter le phishing et la prise de contrôle des comptes. Enfin, le respect des réglementations comme le RGPD impose une bonne gestion des données personnelles et un contrôle strict des accès.

Une approche complète de la sécurité permet donc de protéger efficacement les utilisateurs et l'entreprise contre les menaces informatiques.

Partie B – Correction technique et exercices de complétion

1.Sécurisation de la requête SQL et protection contre le XSS

```
<?php

$conn = mysqli_connect("localhost", "user", "pass", "biblio");

if (!$conn) {

    die("Erreur de connexion : " . mysqli_connect_error());

}


$search = isset($_GET['search']) ? trim($_GET['search']) : ''; // Vérification et nettoyage
```

```
if (!empty($search)) {
    $stmt = mysqli_prepare($conn, "SELECT * FROM livres WHERE titre LIKE CONCAT('%', ?, '%')");
    if ($stmt) {
        mysqli_stmt_bind_param($stmt, "s", $search);
        mysqli_stmt_execute($stmt);
        $result = mysqli_stmt_get_result($stmt);
        while ($row = mysqli_fetch_assoc($result)) {
            // Sécuriser l'affichage pour prévenir le XSS
            echo "<h3>" . htmlspecialchars($row["titre"], ENT_QUOTES, 'UTF-8') . "</h3>";
            echo "<p>" . htmlspecialchars($row["description"], ENT_QUOTES, 'UTF-8') . "</p>";
        }
        mysqli_stmt_close($stmt);
    } else {
        echo "Erreur lors de la préparation de la requête.";
    }
} else {
    echo "Aucun terme de recherche fourni.";
}

mysqli_close($conn);
?>
```

2 . Validation du format du paramètre et de l'email

2.1 - Vérification que $search ne contient que des lettres et des espaces

Ajoutez ce code après la récupération de $search :

```
if (!preg_match('/^[a-zA-ZÀ-ÿ\s]+$/', $search)) {
    die("Le terme de recherche contient des caractères invalides.");
```

}

2.2 - Vérification du format d’une adresse email

Pour vérifier qu’une adresse email est valide, utilisez la fonction filter_var() :

```
$email = $_POST['email'] ?? ''; // Récupération et validation de l'email
if (!filter_var($email, FILTER_VALIDATE_EMAIL)) {
    die("L'adresse email n'est pas valide.");
}
```

3. Réflexion sur la gestion des mots de passe et la sécurité des emails

1. Bonnes pratiques pour la gestion et le stockage sécurisé des mots de passe

La gestion des mots de passe est un élément essentiel de la sécurité des comptes utilisateurs. Voici les meilleures pratiques à adopter :

Stockage sécurisé :

- Ne jamais stocker les mots de passe en clair dans la base de données.
- Utiliser un algorithme de hachage sécurisé comme bcrypt, Argon2 avec un **sel aléatoire** pour éviter les attaques par force brute.

Politiques de mots de passe robustes :

- Exiger un mot de passe d'au moins 12 caractères, incluant des lettres majuscules et minuscules, des chiffres et des caractères spéciaux.
- Mettre en place une vérification de complexité et éviter les mots de passe trop courants.

Gestion du cycle de vie des mots de passe :

- Imposer un changement périodique en cas de compromission.
- Bloquer les tentatives répétées de connexion pour éviter les attaques par brute-force.
- Proposer une authentification à double facteur (2FA) pour renforcer la sécurité.

2. Sécurisation des communications par email

Les emails sont une cible privilégiée des cyberattaques (phishing, usurpation d’identité, etc.), d’où l’importance de renforcer leur sécurité :

Vérification et chiffrement des emails :

- Utiliser des protocoles comme TLS pour chiffrer les emails en transit.
- Vérifier l'authenticité des expéditeurs avec SPF, DKIM et DMARC pour éviter le spoofing.

Gestion des emails contenant des données sensibles :

- Ne jamais envoyer de mots de passe en clair par email.
- Utiliser des solutions de chiffrement de bout en bout pour les emails sensibles.
- Mettre en place une authentification forte (2FA) pour sécuriser l'accès aux boîtes mail.

Prévention contre le phishing :

- Intégrer des alertes pour détecter les emails suspects et bloquer les liens dangereux.
- Sensibiliser les utilisateurs aux attaques de phishing et aux risques d'ingénierie sociale.

3. Importance de la formation des utilisateurs et développeurs

Même avec de bonnes mesures techniques, l'erreur humaine reste un facteur de risque majeur. La sensibilisation et la formation sont donc essentielles.

Formation des utilisateurs :

- Apprendre à reconnaître les emails de phishing et les techniques d'arnaque courantes.
- Encourager l'utilisation de gestionnaires de mots de passe pour éviter les mots de passe faibles ou réutilisés.
- Expliquer l'importance de l'authentification à double facteur et comment l'activer.

Formation des développeurs :

- Les sensibiliser aux vulnérabilités liées à l'authentification (ex : stockage en clair, hachage faible).
- Leur apprendre à implémenter correctement les bonnes pratiques (ex : requêtes préparées, validation des entrées, chiffrement).
- Faire des audits de sécurité et des tests de pénétration réguliers pour détecter les failles.

Partie C – Rapport synthétique et recommandations stratégiques

Rapport de Sécurité et Recommandations Stratégiques

Biblionet

Sommaire

1. **Introduction**
2. **Analyse des vulnérabilités**
    - Injection SQL
    - Cross-Site Scripting (XSS)
    - Gestion des données personnelles et conformité RGPD/CNIL
    - Pratiques de sécurité (mots de passe et emails)
3. **Risques stratégiques**
4. **Correctifs techniques**
5. **Recommandations stratégiques**
6. **Conclusion**

---

1. Introduction

L'application Biblionet a récemment été confrontée à plusieurs failles de sécurité critiques. L'objectif de ce rapport est d'identifier les vulnérabilités, d'évaluer les risques associés et de proposer des mesures correctives et stratégiques pour renforcer la sécurité de l'application.

2. Analyse des vulnérabilités

2.1 Injection SQL

- L'application permet aux utilisateurs de rechercher des livres sans filtrer les entrées, ce qui expose la base de données à des attaques par injection SQL.
- Risques : vol ou suppression de données, prise de contrôle de la base.

2.2 Cross-Site Scripting (XSS)

- L'affichage des résultats de recherche n'est pas protégé contre l'insertion de scripts malveillants.
- Risques : vol de session utilisateur, redirection vers des sites frauduleux.

2.3 Gestion des données personnelles et conformité RGPD/CNIL

- Absence de vérification de l'authenticité des emails et politiques de mots de passe trop faibles.
- Risques : non-conformité aux règles de protection des données, amendes potentielles, perte de confiance des utilisateurs.

2.4 Pratiques de sécurité (mots de passe et emails)

- Mots de passe stockés sans chiffrement adapté.
- Risques : compromission des comptes utilisateurs en cas de fuite de données.

3. Risques stratégiques

Ne pas corriger ces vulnérabilités expose Biblionet à :

- Des cyberattaques menaçant l'intégrité des données.
- Une atteinte à la réputation de l'entreprise.
- Une mise en conformité urgente sous peine de sanctions RGPD.
- Une perte de confiance des utilisateurs et partenaires.

4. Correctifs techniques

- **Requêtes préparées** pour protéger contre l'injection SQL.
- **Utilisation de htmlspecialchars()** pour empêcher les attaques XSS.
- **Validation stricte des entrées utilisateurs** (ex : regex pour empêcher les caractères spéciaux dans les champs sensibles).
- **Stockage des mots de passe avec bcrypt ou Argon2** pour une meilleure sécurité.
- **Authentification à double facteur (2FA)** pour renforcer la protection des comptes.

5. Recommandations stratégiques

5.1 Politiques de sécurité conformes au RGPD

- Mise en place de **procédures d'authentification sécurisées**.
- Chiffrement des données personnelles.
- Accès limité aux informations sensibles.

5.2 Renforcement de la gestion des mots de passe

- Exiger un minimum de **12 caractères** pour les mots de passe.
- Bloquer les tentatives de connexion après plusieurs échecs.
- Imposer le changement régulier des mots de passe compromis.

5.3 Sécurisation des communications par email

- Chiffrement des emails avec **TLS**.
- Implémentation de **SPF, DKIM et DMARC** pour limiter le phishing.
- Sensibilisation des utilisateurs aux attaques d'usurpation d'identité.

5.4 Audits et tests réguliers

- Effectuer des **tests de pénétration** pour détecter les failles.
- Mettre en place une **veille sécurité** pour suivre les nouvelles menaces.
- Sensibiliser régulièrement les employés aux bonnes pratiques.

6. Conclusion

En appliquant ces correctifs et recommandations, Biblionet bénéficiera de :

- **Une meilleure protection des données utilisateurs**.
- **Une réduction significative des risques d'attaques**.
- **Une conformité avec la réglementation (RGPD, CNIL)**.
- **Un renforcement de la confiance des clients et partenaires**.

Il est essentiel que Biblionet adopte une **approche proactive** en matière de cybersécurité afin de garantir la pérennité de son service et de préserver sa réputation.

QCM Technique – Sécurité Informatique

1. Que signifie le terme XSS ?

Réponse : a) Cross-Site Scripting
Justification : Le Cross-Site Scripting (XSS) est une vulnérabilité permettant l'injection de scripts malveillants dans une page web visitée par un utilisateur. Ces scripts peuvent être utilisés pour voler des cookies, détourner des sessions ou injecter du contenu nuisible.

2. Quel est l'objectif principal des requêtes préparées en PHP ?

Réponse : a) Empêcher l'injection SQL
Justification : Les requêtes préparées séparent le code SQL des données utilisateur, empêchant ainsi l'injection SQL. Elles garantissent que les entrées utilisateur ne seront jamais interprétées comme du code SQL.

3. Quelle fonction en PHP est généralement utilisée pour échapper les caractères spéciaux et prévenir les attaques XSS ?

Réponse : b) htmlspecialchars()
Justification : htmlspecialchars() convertit les caractères spéciaux en entités HTML, empêchant ainsi l'exécution de scripts malveillants injectés dans une page web.

4. Que signifie l'acronyme RGPD ?

Réponse : a) Règlement Général sur la Protection des Données
Justification : Le RGPD est un cadre légal européen visant à protéger les données personnelles des citoyens. Il impose des règles strictes sur la collecte, le stockage et le traitement des données.

5. Quel est l'objectif principal d'un pare-feu (firewall) dans un contexte de sécurité réseau ?

Réponse : a) Bloquer le trafic non autorisé
Justification : Un pare-feu filtre les connexions entrantes et sortantes pour empêcher les accès non autorisés aux ressources du réseau.

6. Pourquoi est-il recommandé de hacher les mots de passe avant de les stocker ?

Réponse : a) Pour protéger les mots de passe en cas de fuite de la base de données

Justification : Le hachage empêche le stockage des mots de passe en clair, rendant leur récupération difficile en cas de fuite.

7. Quel protocole est recommandé pour sécuriser l'envoi des emails ?

Réponse : b) SMTPS (ou SMTP avec TLS)
Justification : SMTPS utilise TLS pour chiffrer les communications entre les serveurs de messagerie, protégeant ainsi les emails contre l'interception.

8. Qu'est-ce qu'un WAF (Web Application Firewall) et quel est son rôle ?

Réponse : a) Un dispositif qui surveille et filtre le trafic HTTP pour protéger les applications web
Justification : Un WAF protège les applications web en filtrant les requêtes malveillantes, notamment contre les attaques XSS et SQLi.

9. Que permet de réaliser l'utilisation de htmlspecialchars en PHP ?

Réponse : a) Convertir des caractères spéciaux en entités HTML pour prévenir le XSS
Justification : Cette fonction empêche les scripts injectés d'être exécutés en les rendant inoffensifs.

10. Quel est le rôle principal de la CNIL ?

Réponse : a) Contrôler et garantir la protection des données personnelles en France
Justification : La CNIL veille au respect du RGPD et protège les droits des citoyens en matière de données personnelles.

11. Que signifie le terme "injection SQL" ?

Réponse : a) L'insertion de code malveillant dans une requête SQL pour accéder ou manipuler des données
Justification : L'injection SQL consiste à insérer des commandes SQL non prévues via des champs de saisie pour manipuler la base de données.

12. Quel algorithme est recommandé pour le hachage sécurisé des mots de passe ?

Réponse : c) bcrypt
Justification : bcrypt est conçu pour être résistant aux attaques par force brute grâce à sa lenteur et sa gestion du "salt".

13. Quel protocole assure une communication sécurisée entre un client et un serveur web ?

Réponse : b) HTTPS
Justification : HTTPS utilise TLS/SSL pour chiffrer les communications, garantissant la confidentialité et l'intégrité des données échangées.

14. Quel est le rôle d'un VPN (Virtual Private Network) en cybersécurité ?

Réponse : b) Chiffrer la connexion internet pour sécuriser les échanges de données
Justification : Un VPN crée un tunnel sécurisé entre un utilisateur et un réseau distant pour empêcher l'interception des données.

15. Parmi les principes du "CIA triad", lequel n'en fait pas partie ?

Réponse : d) Efficacité
Justification : La "triade CIA" (Confidentialité, Intégrité, Disponibilité) est un modèle fondamental de la cybersécurité, alors que l'efficacité n'en fait pas partie.

16. Que signifie "Two-Factor Authentication" (2FA) ?

Réponse : a) Utiliser deux méthodes différentes pour vérifier l'identité d'un utilisateur
Justification : La 2FA ajoute une couche de sécurité supplémentaire en exigeant un second facteur, comme un code SMS ou une application d'authentification.

17. Qu'est-ce qu'une vulnérabilité "zero-day" ?

Réponse : b) Une vulnérabilité inconnue des fabricants et sans correctif disponible
Justification : Les failles "zero-day" sont exploitées avant que les développeurs n'aient eu le temps de publier un correctif.

18. Que signifie le terme "phishing" en cybersécurité ?

Réponse : b) La tentative de tromper un utilisateur pour obtenir des informations sensibles via de faux emails ou sites web
Justification : Le phishing exploite la confiance des utilisateurs pour les inciter à divulguer des informations personnelles.

19. Laquelle des propositions décrit le mieux le concept de "Social Engineering" ?

Réponse : b) L’exploitation de la confiance humaine pour obtenir des informations ou accéder à des systèmes
Justification : Le social engineering manipule les individus pour contourner les mesures de sécurité techniques.

20. Quel est le rôle principal d’un IDS (Intrusion Detection System) ?

Réponse : b) Surveiller et détecter des comportements anormaux ou des tentatives d’intrusion dans un réseau
Justification : Un IDS analyse le trafic réseau pour identifier et signaler les activités suspectes pouvant indiquer une tentative d'intrusion.