PROJET SSI

DANY GHASSAN
IBRAHIM EL-OUARDI
KAYSS LELEU

Sommaire :

Depot git : https://github.com/danyGhassan/SSI

I-Introduction

L'évolution rapide de la technologie a conduit à une dépendance croissante à l'égard des services en ligne et des applications web, soulignant ainsi l'importance cruciale de la sécurité des infrastructures et des réseaux informatiques. Dans ce contexte, la conception et la sécurisation des sites web revêtent une importance particulière, car ils constituent souvent la première ligne de défense contre les attaques cybernétiques.

Le présent rapport se concentre sur le processus de conception et de sécurisation d'un site web, ainsi que sur la démonstration de sa résilience face aux attaques par injection SQL. Ce projet s'inscrit dans le cadre de la première année d'études en informatique, offrant ainsi une opportunité d'appliquer les concepts théoriques abordés durant cette année.

L'objectif principal de ce projet était de créer un site web sécurisé en mettant en œuvre des mesures de sécurité robustes, notamment la validation des entrées utilisateur et la prévention des attaques par injection SQL. En outre, il visait à démontrer l'efficacité de ces mesures en effectuant des tentatives d'injections SQL contrôlées, mettant ainsi en lumière l'importance de la sécurité dans le développement web.

Ce rapport détaillera le processus de réalisation du projet, en mettant en évidence les étapes clés de conception, de développement et de sécurisation du site web. Il examinera également les difficultés rencontrées et les leçons apprises tout au long du processus, offrant ainsi un aperçu complet des défis et des solutions liés à la sécurisation des infrastructures et réseaux informatiques.

Enfin, une démonstration du site web sécurisé sera présentée, accompagnée de captures d'écran annotées, illustrant ainsi la mise en pratique des mesures de sécurité implémentées et leur efficacité face aux tentatives d'injections SQL.

II-Prérequis

La réalisation d'un projet de sécurisation de site web et de démonstration d'injections SQL requiert une base solide de connaissances et de compétences dans plusieurs domaines clés de l'informatique. Avant de se lancer dans ce projet, il est essentiel de maîtriser les éléments suivants :

1. **Langages de programmation web** : Une compréhension approfondie des langages de programmation web tels que HTML, CSS et Golang est indispensable. Ces langages sont utilisés pour créer l'interface utilisateur et le contenu visuel du site web.(On a choisi d’utiliser le Golang on a décidé de reprendre un ancien projet réalisé pendant notre formation de première année tout en le sécurisant)
2. **Bases de données relationnelles** : Une connaissance pratique des bases de données relationnelles est nécessaire pour concevoir et gérer la structure de la base de données du site web. Dans ce projet, nous avons utilisé SQLite comme système de gestion de base de données.
3. **Sécurité informatique** : Une compréhension des principes fondamentaux de la sécurité informatique est essentielle pour concevoir des mesures de sécurité efficaces contre les attaques en ligne. Cela inclut la familiarité avec les vulnérabilités courantes telles que les injections SQL et les méthodes pour les prévenir.

4. **Validation des entrées utilisateur** : La capacité à mettre en œuvre des techniques de validation des entrées utilisateur est cruciale pour prévenir les attaques par injection SQL. Cela comprend la validation des formulaires et la désinfection des données entrantes pour éliminer les caractères dangereux.
5. **Gestion de projet** : Une compétence en gestion de projet est utile pour planifier et organiser les différentes étapes du projet, en définissant des objectifs clairs et en respectant les délais impartis.

III- Objectifs

Le projet de sécurisation d'un site web et de démonstration d'injections SQL poursuit plusieurs objectifs clés, visant à combiner théorie et pratique dans le domaine de l'informatique et de la sécurité des systèmes d'information. Les principaux objectifs du projet sont les suivants :

6. **Conception d'un site web sécurisé** : Le premier objectif est de reprendre un site web fonctionnel qui a été réalisé au préalable dans un ancien projet, et ensuite le rendre attrayant et sécurisé. Cela implique la création d'une interface utilisateur conviviale, la mise en place d'une architecture solide et la sécurisation des données utilisateur contre les attaques potentielles.
7. **Implémentation de mesures de sécurité** : Un objectif essentiel est d'implémenter des mesures de sécurité robustes pour protéger le site web contre les menaces potentielles, notamment les attaques par injection SQL. Cela comprend la validation des entrées utilisateur, la désinfection des données entrantes et d'autres pratiques de sécurité recommandées.
8. **Démonstration de la résilience aux injections SQL** : Un objectif clé est de démontrer l'efficacité des mesures de sécurité mises en place en effectuant des tentatives d'injections SQL contrôlées. L'objectif est de montrer que le site web est capable de détecter et de bloquer les tentatives d'attaques, préservant ainsi l'intégrité de la base de données.
9. **Application des concepts théoriques** : Ce projet vise à appliquer les concepts théoriques appris en cours, en mettant l'accent sur la sécurisation des sites web et la prévention des attaques informatiques. Il offre ainsi une opportunité d'approfondir la compréhension des principes de sécurité informatique dans un contexte pratique.
10. **Apprentissage par l'expérience** : Enfin, un objectif important est de permettre aux étudiants d'apprendre par l'expérience en travaillant sur un projet concret. En affrontant des défis réels liés à la sécurité des systèmes d'information, les étudiants développent leurs compétences techniques et leur sensibilisation à la sécurité informatique.

IV-Réalisation

La réalisation du projet de sécurisation d'un site web et de démonstration d'injections SQL s'est déroulée en plusieurs étapes clés, couvrant la conception, le développement et la mise en œuvre des mesures de sécurité. Voici un aperçu détaillé de chaque étape :

11. **Conception et développement du site web** :

    - Cette étape a débuté par la conception de l'architecture du site web, en définissant les fonctionnalités principales et l'organisation générale de l'interface utilisateur.
    - Le développement du site web a ensuite été réalisé en utilisant des langages de programmation web tels que HTML, CSS et Golang. L'objectif était de créer une interface utilisateur attrayante et conviviale, tout en assurant la compatibilité avec différents navigateurs et appareils.

12. **Configuration de la base de données** :

    - Une base de données SQLITE a été configurée pour stocker les informations du site web de manière sécurisée. La structure de la base de données a été conçue en fonction des besoins fonctionnels du site, en tenant compte des relations entre les différentes entités.

13. **Implémentation des mesures de sécurité** :

    - Des mesures de sécurité robustes ont été mises en place pour protéger le site web contre les attaques par injection SQL et autres menaces potentielles. Cela comprenait la validation rigoureuse des entrées utilisateur, la désinfection des données entrantes et l'utilisation de requêtes paramétrées pour interagir avec la base de données.
    - Des mécanismes de gestion des sessions et d'authentification ont également été implémentés pour contrôler l'accès aux fonctionnalités sensibles du site web.

14. **Implémentation des mesures contre le brut force :**
    Nous avons mis en place un programme qui empêche le brut force, le client possède 5 tentatives pour se connecter. Si il échoue, il est bloquer du serveur et ne peut donc plus se connecter.

15. **Implémentation d’un hash pour les mot de passe :**
    Nous avons hashé(sha256) le mot de passe qui permet d’accéder au site avec un sel, pour renforcer la sécurité du site.

16. **Les paths sont bloqué :**
    Le seul moyen d’accéder aux autres pages est d’être connecté

2. **Démonstration des injections SQL** :

    - Une série de tentatives d'injections SQL contrôlées ont été effectuées pour tester la résilience du site web aux attaques. Des scénarios d'attaques courants ont été simulés, tels que l'insertion de code SQL malveillant dans les champs de formulaire.
    - Les résultats de ces tentatives ont été analysés pour évaluer l'efficacité des mesures de sécurité mises en place et identifier d'éventuelles vulnérabilités à corriger.

En combinant ces étapes, le projet a abouti à la création d'un site web sécurisé, résistant aux attaques par injection SQL. La réalisation de ce projet a permis d'appliquer les concepts théoriques appris en cours d'infrastructures et réseaux dans un contexte pratique, offrant ainsi une expérience enrichissante. Nous précisons également que toutes ses implémentations ont été réaliser a l'aide de nos connaissances, ce sont seulement des programmes crée par nous.

V-Difficultés rencontrées

La réalisation du projet n'a pas été exempte de défis, certains rencontrés ayant nécessité des efforts supplémentaires pour les surmonter. Les principales difficultés rencontrées comprenaient :

1. **Équilibre entre sécurité et fonctionnalité** :
    - Trouver un équilibre entre la sécurisation du site web et sa fonctionnalité utilisateur s'est avéré être un défi majeur. Certaines mesures de sécurité strictes pouvaient interférer avec l'expérience utilisateur, nécessitant ainsi des ajustements pour concilier sécurité et convivialité.

2. **Complexité des injections SQL** :
    - Comprendre et contrer les différentes formes d'attaques par injection SQL a été une tâche ardue. Les attaquants peuvent utiliser diverses techniques pour contourner les mesures de sécurité, nécessitant une vigilance constante et une compréhension approfondie des vulnérabilités potentielles.

3. **Tests et débogage** :

• La réalisation de tests d'injections SQL contrôlées a exigé une planification minutieuse et une exécution précise. Identifier les vulnérabilités et corriger les failles de sécurité détectées a nécessité du temps et des efforts supplémentaires pour garantir l'intégrité du site web.

4. **Hébergement en ligne non réalisable** :

    - Malgré nos efforts, nous n'avons pas pu héberger le site web sécurisé en ligne en raison de contraintes techniques et de ressources limitées. Cette difficulté a limité notre capacité à démontrer le site web en action et à tester sa résilience dans un environnement réel.Mais malgré tout nous avons trouvé trouvé un moyen de pouvoir héberger le site sans avoir à payer un nom de domaine, cependant le site ne supporte pas le Golang, donc par manque de connaissance des autres langages en web et ainsi de temps, nous avons décidé de laisser comme tel.

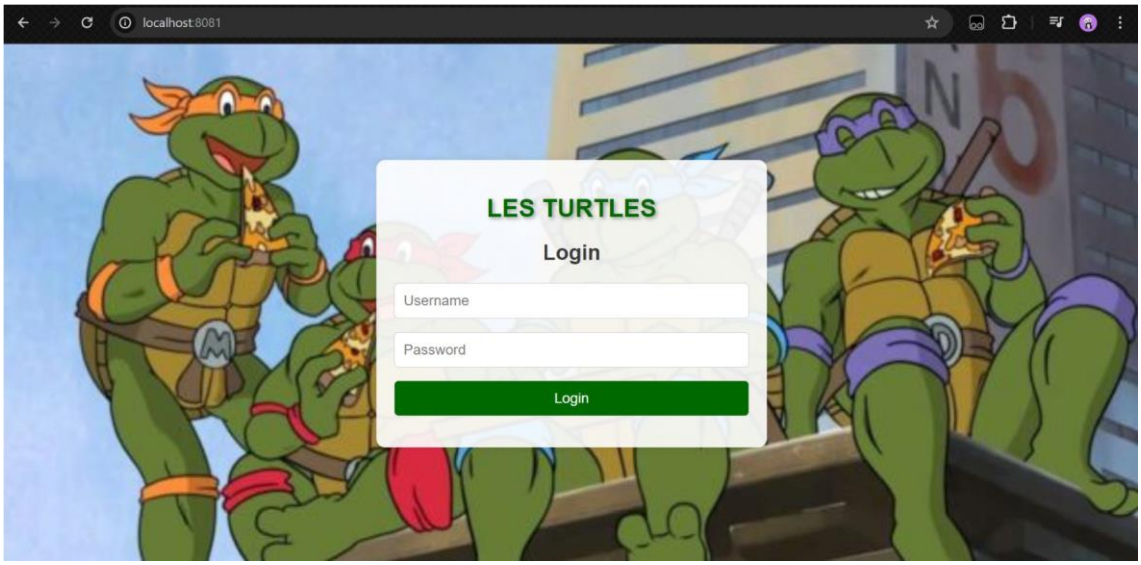
Ces difficultés ont représenté des obstacles significatifs tout au long du projet, mais elles ont également fourni des occasions d'apprentissage précieuses. En les surmontant, nous avons acquis une compréhension plus approfondie des défis associés à la sécurisation des sites web et renforcé nos compétences dans ce domaine en constante évolution.

VI-Démo avec captures d'écran et explications

Une démonstration du site web sécurisé a été réalisée pour illustrer les mesures de sécurité mises en place et leur efficacité face aux tentatives d'injections SQL. Voici les principales étapes de la démonstration, accompagnées de captures d'écran annotées :

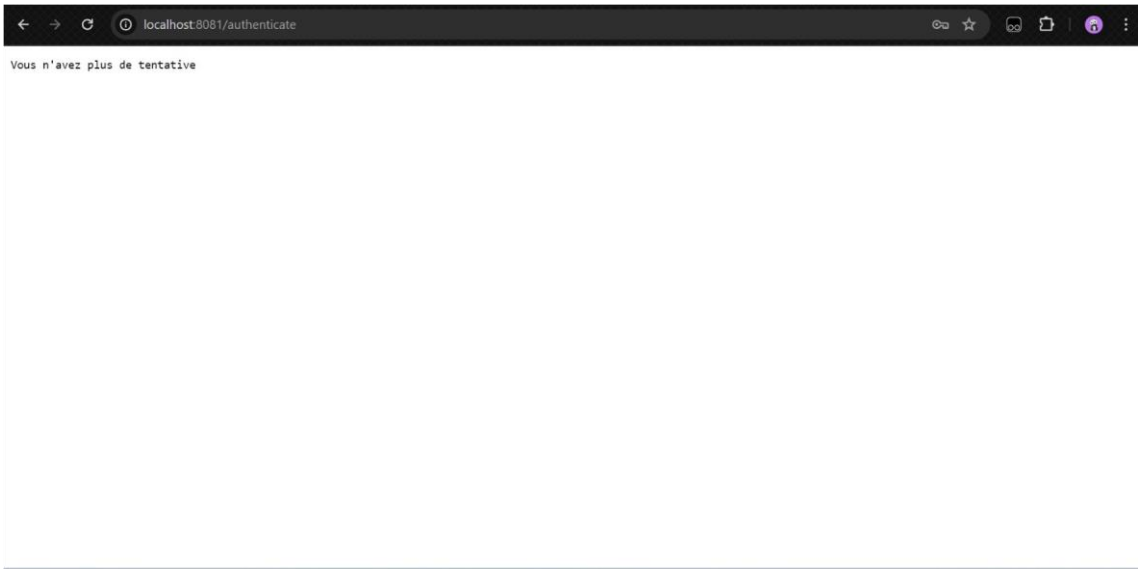
1. **Page de connexion sécurisée** :

    - La première étape consiste à accéder à la page de connexion du site web. Cette page demande à l'utilisateur de saisir son identifiant et son mot de passe pour accéder à son compte.



2. **Validation des entrées utilisateur** :

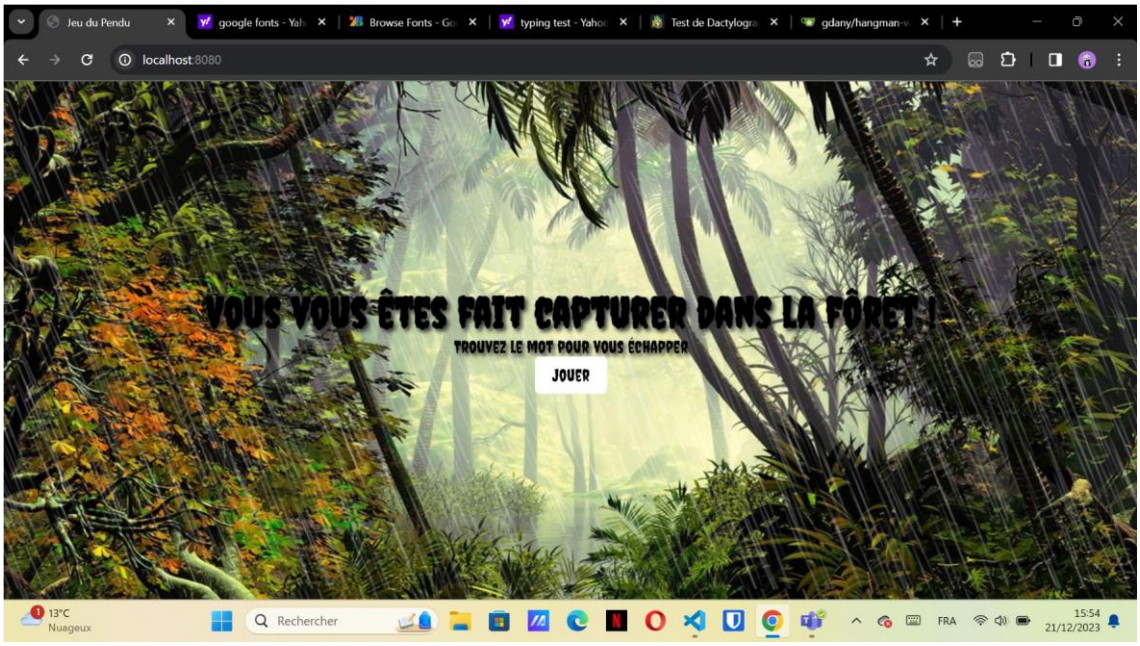
Lorsque l'utilisateur saisit ses informations de connexion, le site web valide les entrées utilisateur pour prévenir les attaques par injection SQL. Toute tentative d'injection de code SQL malveillant est détectée et bloquée .



3. **Accès au tableau de bord sécurisé** :

- Une fois les informations de connexion validées, l'utilisateur est redirigé vers le tableau de bord sécurisé du site web. Ce tableau de bord offre un accès aux fonctionnalités principales de l'application.

-



4. **Tests d'injections SQL contrôlées** :

    - Pour démontrer la résilience du site web aux attaques par injection SQL, des tests d'injections contrôlées ont été effectués. Différents scénarios d'attaques ont été simulés, tels que l'insertion de code SQL dans les champs de formulaire.

```go
func containsForbiddenChars(input string) bool {
    for _, char := range []string{"'", `\`, "#", ";", ")", ",", "\"", "\b", "\n", "\r", "\t", "\\", "%", "_", "-", "*", "$"} {
        if strings.Contains(input, char) {
            return true
        }
    }
    return false
}
```

En suivant ces étapes, la démonstration met en lumière l'efficacité des mesures de sécurité mises en place pour prévenir les attaques par injection SQL.

VII-Clarté des explications et des étapes

La clarté des explications et des étapes tout au long du projet est essentielle pour assurer une compréhension approfondie des mesures de sécurité mises en place et des résultats obtenus. Voici comment cette clarté a été assurée :

1. **Explications détaillées des mesures de sécurité** :

    Chaque mesure de sécurité implémentée, telle que la validation des entrées utilisateur et la désinfection des données entrantes, a été expliquée en détail. Les principes sousjacents à ces mesures ont été clarifiés pour permettre une compréhension approfondie de leur fonctionnement.

    - Les explications ont été fournies de manière claire et concise, évitant l'utilisation de jargon technique excessif pour faciliter la compréhension même pour les lecteurs moins familiers avec les concepts de sécurité informatique.

2. **Décomposition des étapes de démonstration** :

- Les différentes étapes de la démonstration du site web sécurisé ont été clairement décomposées pour faciliter leur compréhension. Chaque étape a été présentée de manière séquentielle, avec des explications détaillées des actions entreprises à chaque étape.
- Les captures d'écran annotées ont été utilisées pour illustrer visuellement chaque étape de la démonstration, offrant ainsi une référence visuelle claire pour accompagner les explications textuelles.

3. **Utilisation d'exemples concrets** :

- Des exemples concrets ont été utilisés pour illustrer les concepts abordés et les mesures de sécurité mises en place. Ces exemples ont permis aux lecteurs de mieux visualiser l'application pratique des principes théoriques discutés dans le rapport.
- Les scénarios d'attaques par injection SQL contrôlées ont été décrits en détail, avec des explications sur les actions de l'attaquant et les réponses du système de sécurité du site web.

VIII-Fonctionnalités implémentées

Le site web sécurisé comprend plusieurs fonctionnalités clés, conçues pour offrir une expérience utilisateur enrichissante tout en garantissant la sécurité des données. Voici un aperçu détaillé des fonctionnalités implémentées :

1. **Système d'authentification sécurisé** :

- Un système d'authentification robuste a été mis en place pour permettre aux utilisateurs de se connecter de manière sécurisée au site web. Les informations d'identification des utilisateurs sont vérifiées à l'aide de méthodes cryptographiques et de contrôles de sécurité rigoureux pour prévenir les attaques d'usurpation d'identité.

2. **Gestion des utilisateurs** :

- Le site web propose des fonctionnalités de gestion des utilisateurs, permettant aux administrateurs d'ajouter, de modifier et de supprimer des comptes utilisateur. Ces fonctionnalités sont soumises à des autorisations strictes pour garantir l'intégrité et la confidentialité des données utilisateur.

3. **Validation des formulaires** :

- Tous les formulaires du site web sont soumis à une validation rigoureuse pour prévenir les attaques par injection SQL et d'autres types d'attaques basées sur les entrées utilisateur. Les données saisies par les utilisateurs sont vérifiées et nettoyées avant d'être traitées par le système.

4. **Protection contre les attaques par injection SQL** :

- Des mesures de sécurité avancées ont été mises en place pour protéger le site web contre les attaques par injection SQL. Les requêtes SQL sont paramétrées et les données entrantes sont désinfectées pour éliminer tout code malveillant potentiel, réduisant ainsi le risque de compromission de la base de données.

-

5. **Journalisation des activités** :

    - Le site web enregistre les activités des utilisateurs, notamment les tentatives de connexion, les modifications de profil et les actions administratives. Ces journaux d'activité sont essentiels pour la détection des comportements suspects et la résolution des incidents de sécurité. Toutes ses fonctionnalité sont permises avec Azur, mais c'est seulement si le site est hébérgé.

En implémentant ces fonctionnalités, le site web sécurisé offre une plateforme robuste et sécurisée pour ses utilisateurs, garantissant ainsi la confidentialité, l'intégrité et la disponibilité des données. Ces fonctionnalités sont conçues pour répondre aux exigences de sécurité les plus strictes tout en offrant une expérience utilisateur fluide et intuitive.

IX-Conclusion

Le projet de sécurisation d'un site web et de démonstration d'injections SQL a été une expérience enrichissante qui a permis d'approfondir notre compréhension des concepts de sécurité informatique et de leur application pratique. En résumé, plusieurs points clés peuvent être mis en évidence :

1. **Importance de la sécurité informatique** :

    - Ce projet a souligné l'importance cruciale de la sécurité informatique dans la conception et le développement de sites web. La protection contre les attaques par injection SQL et autres menaces en ligne est essentielle pour garantir l'intégrité et la confidentialité des données.
2. **Efficacité des mesures de sécurité** :

    - Les mesures de sécurité mises en place, telles que la validation des entrées utilisateur et la désinfection des données entrantes, se sont avérées efficaces pour prévenir les attaques par injection SQL. Les tests d'injections contrôlées ont confirmé la robustesse du système de sécurité du site web.
3. **Complexité des défis** :

    - La réalisation de ce projet a également mis en lumière la complexité des défis associés à la sécurisation des sites web. Trouver un équilibre entre sécurité et fonctionnalité, ainsi que la détection et la prévention des attaques, sont des tâches qui exigent une expertise technique et une vigilance constante.
4. **Opportunités d'apprentissage** :

    Ce projet a offert de nombreuses opportunités d'apprentissage, permettant d'acquérir de nouvelles compétences techniques et de développer une sensibilisation accrue à la sécurité informatique. Les défis rencontrés ont été des occasions d'amélioration et de renforcement des connaissances.

En conclusion, ce projet a été une étape importante dans notre parcours d'apprentissage en informatique, nous fournissant une expérience pratique précieuse dans le domaine de la sécurisation des infrastructures et réseaux informatiques. Les leçons apprises et les compétences acquises seront

précieuses dans notre parcours professionnel futur, en nous préparant à relever les défis de la sécurité informatique dans un monde numérique en constante évolution.

Ibrahim EL-OUARDI
Dany GHASSAN ALFONS
Kayss LELEU

•