Rapport d’audit de l’application « Shop »

Table des matières

[Introduction 2](#_Toc114040681)

[Authentification 3](#_Toc114040682)

[Description 3](#_Toc114040683)

[Éléments audités 3](#_Toc114040684)

[Https : 3](#_Toc114040685)

[Login : 3](#_Toc114040686)

[Hash avec sel : 4](#_Toc114040687)

[Restriction : 4](#_Toc114040688)

[Corrections à apporter 5](#_Toc114040689)

[Https : 5](#_Toc114040690)

[Login : 6](#_Toc114040691)

[Hash avec sel : 6](#_Toc114040692)

[Restriction : 7](#_Toc114040693)

# Introduction

Ce document regroupe les informations sur les menaces web examinées du site "Shop" en les décrivant et en mettant en exergue les éléments du code y faisant référence. Si besoin, les étapes nécessaires à la correction ont été décrites précisément

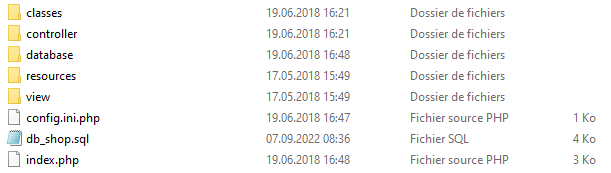
# Authentification

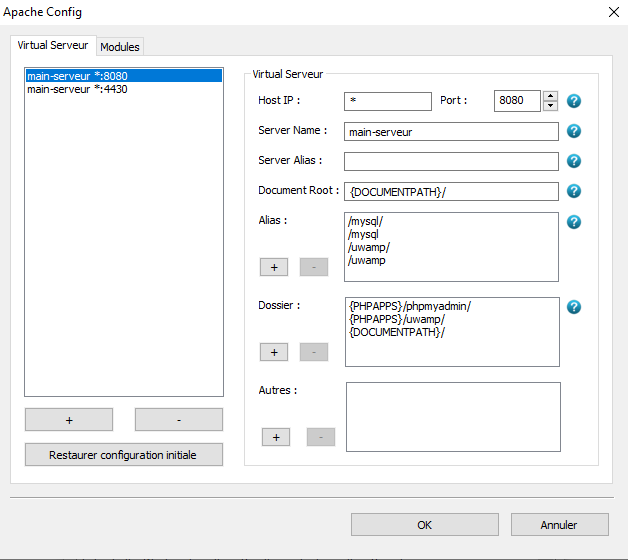
## Description

L’authentification est un processus permettant d’identifier une personne sur un réseau et protéger ses données personnelles. Il est donc important de sécuriser les données lors de la création d’un profil.

## Éléments audités

Https :

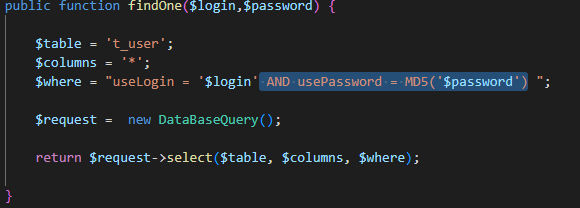
Le site est actuellement en http. Et donc, par conséquent, pas protégé. Si vous avez une ancienne version de Uwamp, il y de grandes chances que les ports de Apache ne soient pas ceux de base.   




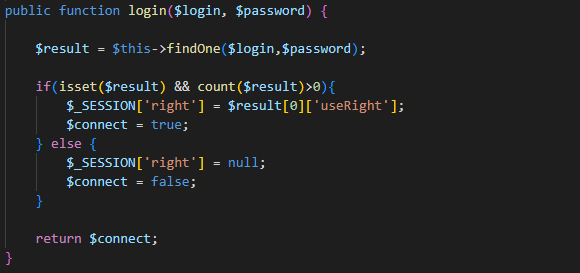
Login :

Les login Admin et Utilisateur n’ont pas un mot de passe sécurisé, il faudrait les changer.

Hash avec sel :

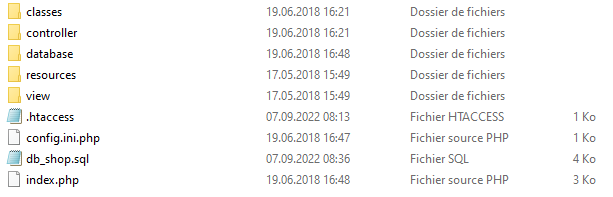
Les mots de passe qui sont stockés en base de données sont encryptés en MD5 et ne sont pas hashé avec un sel.  


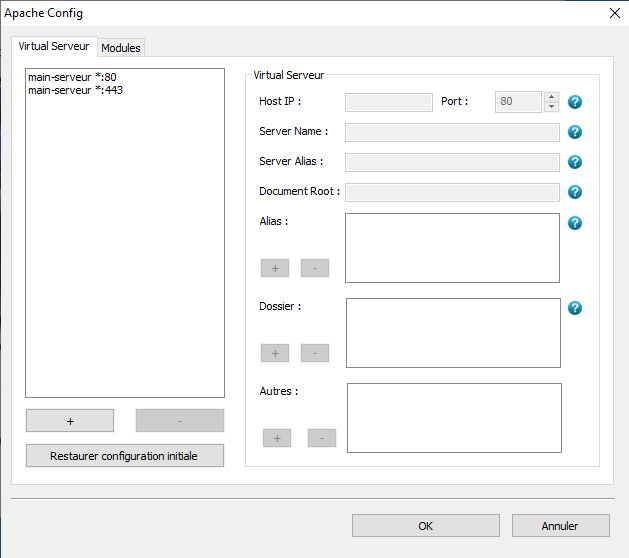
Restriction :

Il n’y a aucune restriction sécuritaire pour les essais de connexion 

## Corrections à apporter

### Https :

Pour rediriger une requête http nous devons créer un fichier .htaccess qui va permettre une redirection automatique en https. A la racine du site, il faut ajouter un fichier nommé « .htaccess ». Ce fichier contient les éléments suivants :   
Une image contenant texte

Description générée automatiquement  
Si les ports sont différents, il faut les modifier. 

Login :

Les mots de passe ne sont actuellement pas sécurisés, il faudrait les changer par un mot de passe plus long et plus sécurisé. Les normes pour avoir un mot de passe sûr et sécurisé, nous vous conseillons la norme suivante :

* Longueur du mot de passe minimum : 16 caractères
* Utiliser des majuscules et minuscules
* Utiliser les chiffres
* Utiliser des caractères spéciaux tel que : \*&@#

Admin : .Admin- -> j@1m3LeSC@Rr07tesV3rtEs  
Utilisateur : .User- -> M0nCh@73stM0R7¢e50iR

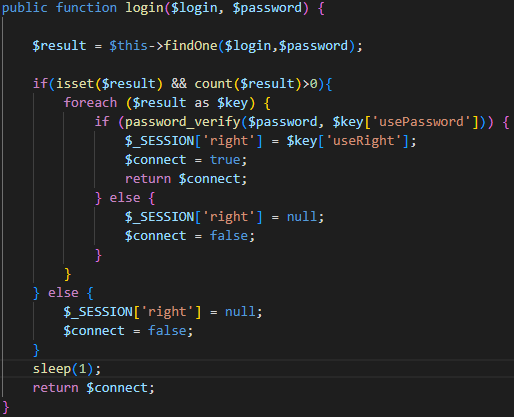
Hash avec sel :

Pour stocker un mot de passe sécurisé en base de données, on peut utiliser la fonction PHP de base, qui hash et ajoute du sel automatiquement :

   
Et on change le code pour qu’à chaque connexion, il vérifie le mot de passe :  
Une image contenant texte

Description générée automatiquement

### Restriction :

Pour la restriction, nous avons ajouté la fonction « sleep() » qui met en pause la lecture du code pendant x secondes. Cette restriction permet de ralentir considérablement les attaques par brute-force.   
  
On peut aussi mettre un compteur dans la variable $\_SESSION et après tant de tentatives de connexion, on empêche l’utilisateur de se connecter pendant un temps imparti et on envoie un mail au possesseur du compte pour lui en informer.