Atelier N°1 : Vulnérabilité XSS

Le cross-site Scripting (abrégé XSS) est un type de faille de sécurité des sites web permettant d'injecter du contenu dans une page, provoquant ainsi des actions sur les navigateurs web visitant la page. Les possibilités des XSS sont très larges puisque l'attaquant peut utiliser tous les langages pris en charge par le navigateur (JavaScript, Java, Flash…) et de nouvelles possibilités sont régulièrement découvertes notamment avec l'arrivée de nouvelles technologies comme HTML5. Il est par exemple possible de rediriger vers un autre site pour de l'hameçonnage ou encore de voler la session en récupérant les cookies.

Les XSS sont donc des injections de code qui vont avoir un impact sur le navigateur de l'utilisateur lors de leur exploitation. L'une des exploitations des XSS les plus utiles pour un pirate est l'injection de code JavaScript dans la page HTML afin de voler le cookie de session de la victime, permettant donc de se connecter à l'application web en se faisant passer pour la victime. Cependant il est possible de faire autre chose en exploitant une XSS comme modifier le contenu d'une page HTML ou bien prendre le contrôle du navigateur de la victime.

→L'objectif de ce TP est d'exécuter avec succès l'attaque de sécurité

Les étapes à suivre :

1-Préparation de l'environnement :

Téléchargement et Installation de XAMPP
https://www.apachefriends.org/fr/download.html

Téléchargement de DVWA https://sourceforge.net/projects/dvwa.mirror/

2- Installation de DVWA :

La DVWA, ou en entier la **Damn Vulnerable Web App** est une application permettant de tester les failles de sécurité. Il s'adresse aux personnes qui souhaitent pratiquer les tests d'intrusion de **manière légale** en utilisant une **cible légale** . Commencer avec la DVWA est l'une des meilleures façons de commencer à apprendre le piratage éthique juridique, l'application est parfaitement adaptée aux utilisateurs de différents niveaux.

L'application est construite avec PHP et MySQL, un duo classique. Qu'est-ce que cela signifie pour quelqu'un qui souhaite apprendre les tests d'intrusion ? Que l'application est facile à installer sur différents systèmes d'exploitation, car PHP et MySQL fonctionnent presque partout. De plus, le fait qu'il soit construit avec PHP signifie qu'il sera plus facile de comprendre les morceaux de code du DVWA. Comme l'application a de nombreux exemples de différentes vulnérabilités (plus à ce sujet plus tard) qui sont implémentées en PHP. PHP n'est pas le langage de programmation le plus difficile à comprendre, il existe également diverses ressources qui aideraient à comprendre le code PHP.

Ses objectifs principaux sont :

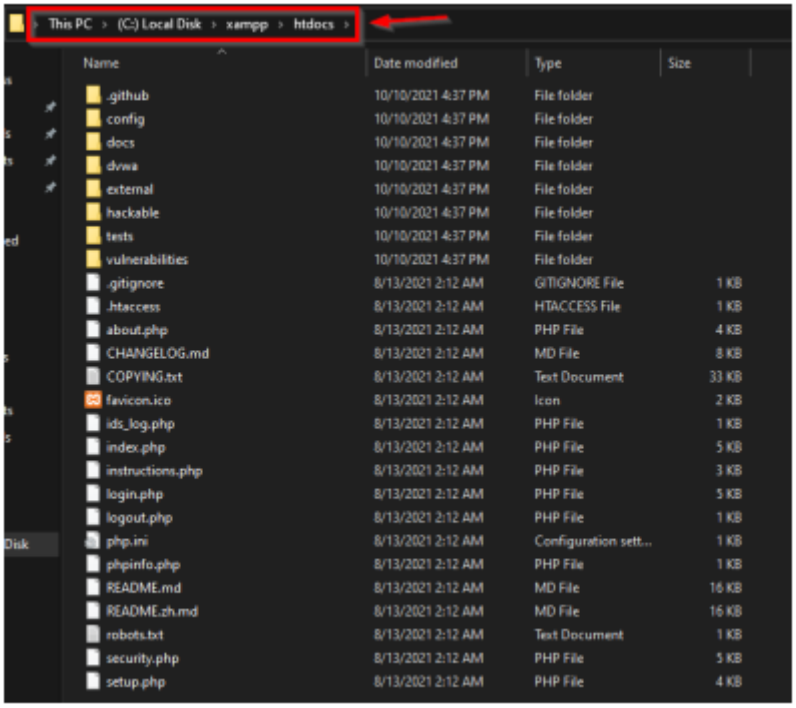
• Aider les professionnels de la sécurité à tester leurs compétences et leurs outils dans un environnement légal.

• Aider les développeurs web à mieux comprendre les processus de sécurisation des applications web.

• Aider les étudiants et les enseignants à apprendre la sécurité des applications web dans un environnement de classe contrôlé. DVWA met en pratique certaines des vulnérabilités web les plus courantes, avec différents niveaux de difficulté, à l'aide d'une interface simple et directe.

Après l’installation, Le fichier DVWA téléchargé sera un fichier zip. Vous devez donc extraire ce fichier.

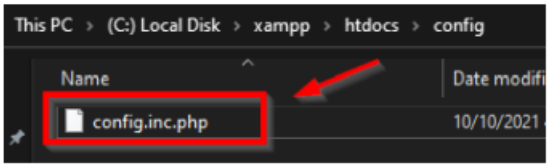
Après avoir installé le serveur XAMPP sur votre ordinateur, veuillez copier le dossier de DVWA dans le répertoire "htdocs" du serveur XAMPP



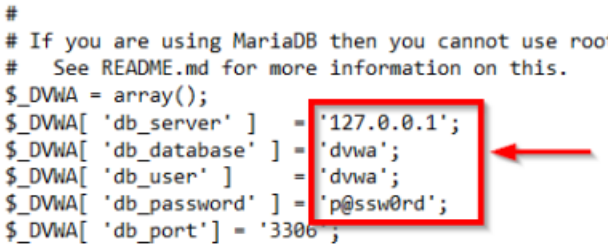
Ouvrez le dossier Config à partir duquel vous pourrez configurer le DVWA et modifier le nom d'utilisateur et le mot de passe qui lui sont associés. Ici, si vous

ne voulez pas changer le nom d'utilisateur et le mot de passe, vous pouvez aussi aller dans ce dossier pour voir le mot de passe

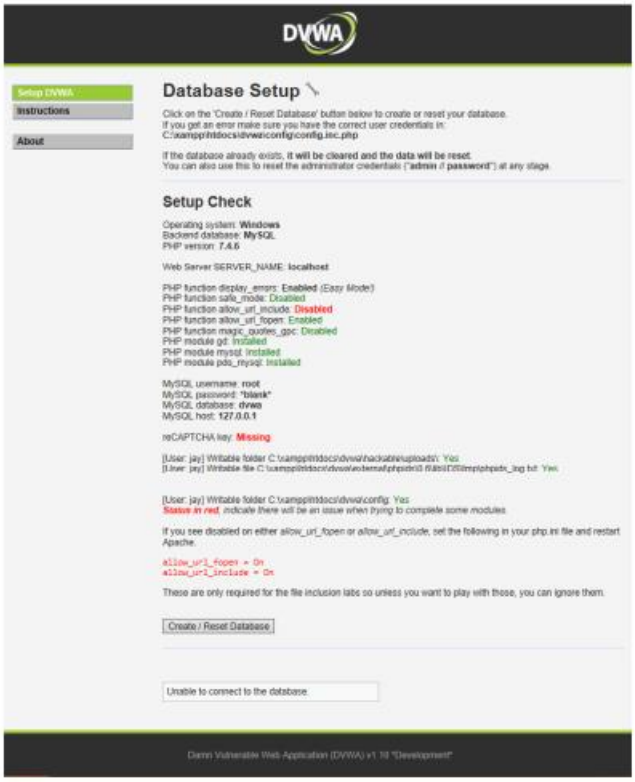
Après avoir ouvert le dossier, un fichier s'affiche devant vous, vous devez d'abord le renommer. Le format (.dist) doit être supprimé. Pour que votre navigateur puisse lire le fichier



Ensuite, vous devez modifier ce fichier en l'ouvrant dans un bloc-notes, puis vous pouvez changer le nom d'utilisateur et le mot de passe pour vous connecter.



Maintenant allez dans votre navigateur web et tapez localhost/dvwa et vous serez présenté avec la page par défaut de dvwa comme ceci :

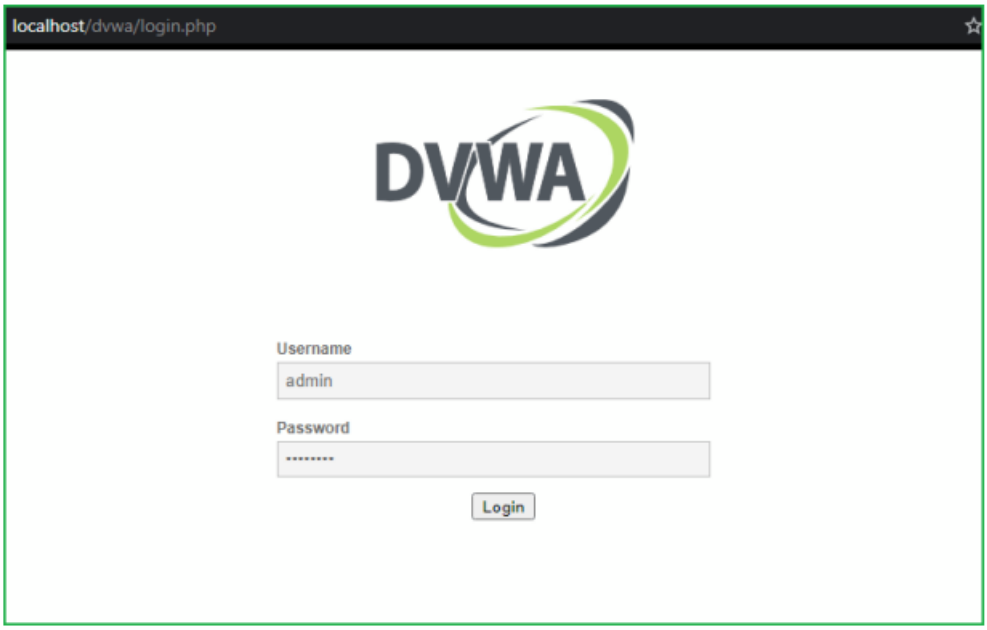


Maintenant cliquez sur Create/reseatDatabase et vous serez redirigé vers la page localhost/dvwa/login.php comme ceci :

Une fois que vous aurez entré votre nom d'utilisateur et votre mot de passe, vous serez redirigé vers localhost/dvwa/index.php comme ceci :

Connexion à l'interface DVWA

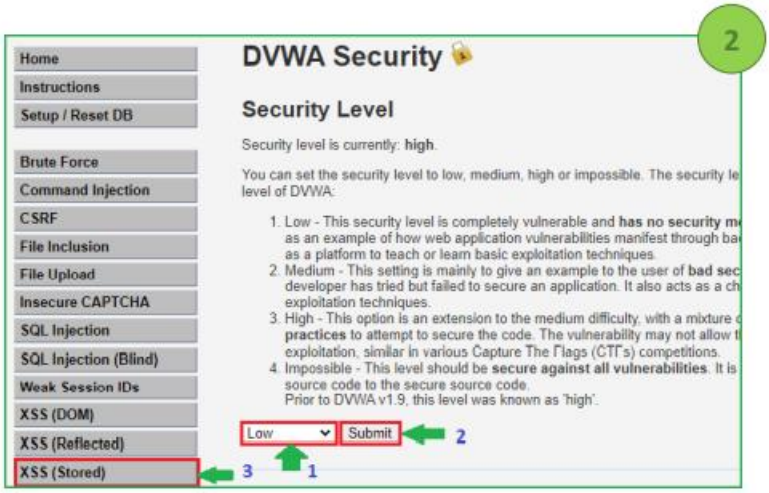
Tout d'abord, connectez-vous à votre application web DVWA avec l'identifiant par défaut admin : password ou un autre identifiant que vous avez défini.



Etape : Changement de niveau de sécurité

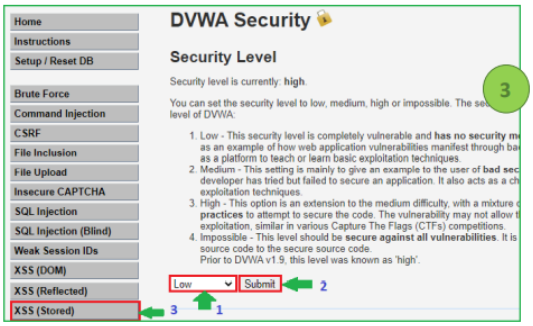
Nous commencerons par un niveau faible et passerons progressivement à un niveau élevé. Cliquez sur la sécurité DVWA dans le volet de gauche pour changer la difficulté en faible.

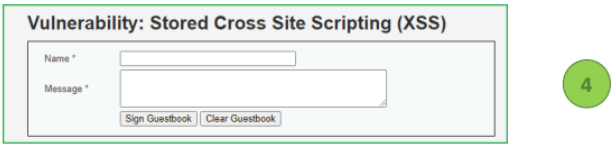
Sélectionnez le niveau de sécurité à faible et soumettez pour soumettre la demande. Cliquez ensuite sur XSS (Stored) dans le volet de gauche pour sélectionner la vulnérabilité XSS stockée car nous allons nous entraîner à l'attaque XSS stockée.



Etape : l'interface du formulaire

Après avoir cliqué sur le bouton XSS (Stored), nous pouvons voir qu'il y a deux champs : Name et Message.

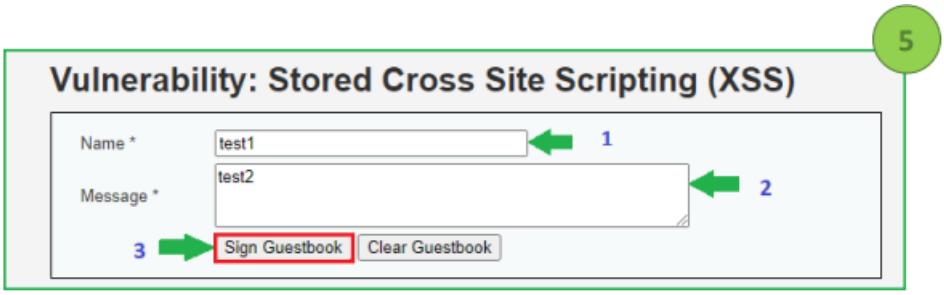




Etape : Test du formulaire

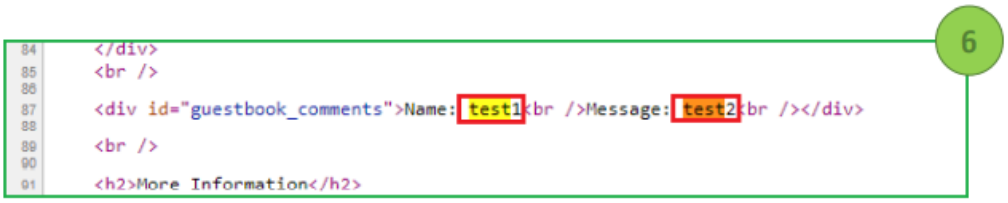
Saisissons une chaîne unique pour vérifier si elle s'affiche ou non dans la fenêtre du navigateur. Dans mon cas, j'ai saisi test1 et test2 dans les champs

Nom et Message respectivement. Ensuite, cliquez sur Signer le livre d'or pour soumettre la demande.



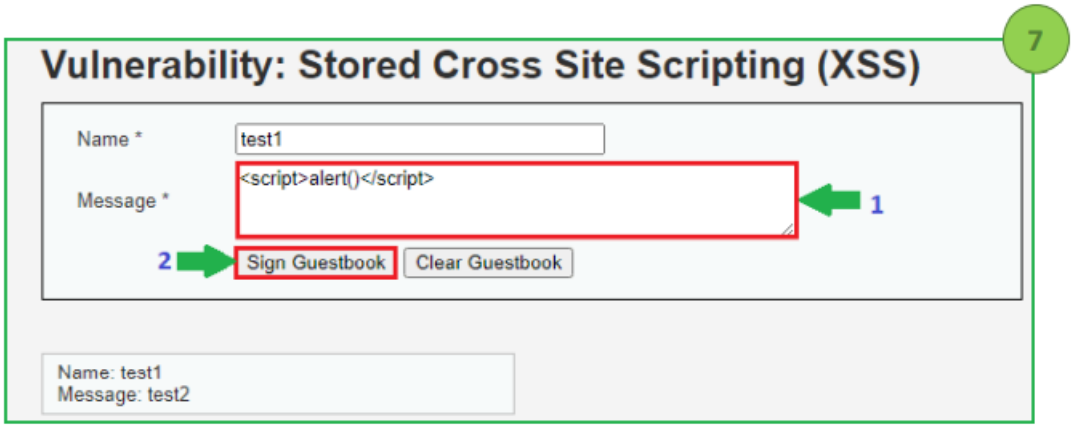
Etape : Vérification de la chaine unique dans le code source

Dès que notre demande est soumise, l'étape suivante consiste à vérifier la source de la page pour savoir si notre chaîne unique est reflétée ou non. En appuyant sur CTRL+U pour vérifier la source de la page, puis en recherchant la chaîne de test, nous avons constaté que test1 et test2 se reflètent tous deux. Puisque les deux se reflètent dans le navigateur, ces deux champs peuvent être vulnérables à une attaque XSS stockée



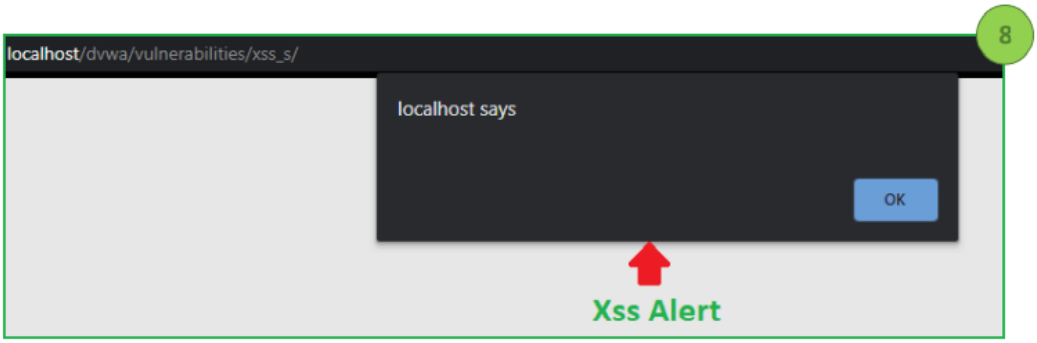
Etape : Envoie de la chaine XSS

Maintenant, notre dernière étape consiste à envoyer une charge utile XSS dans l'un de ces deux champs de saisie. J'utilise une charge utile XSS très basique dans le champ Message. Cliquez sur Signer le livre d'or pour soumettre le message. Si ce site est vulnérable à la vulnérabilité XSS stockée, nous obtiendrons un popup lorsque nous rafraîchirons cette page.



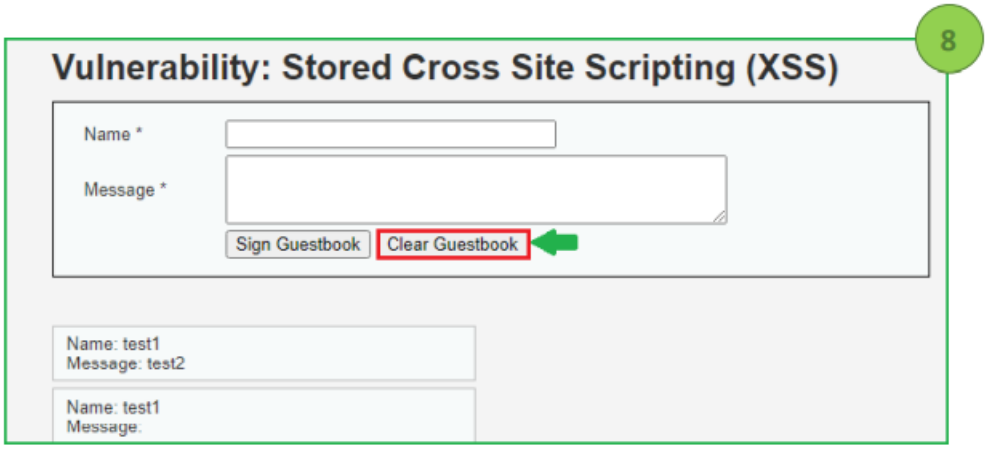
Etape : Résultat soumis

Lorsque j'ai actualisé cette même page, j'ai obtenu une boîte d'alerte XSS. Cette boîte confirme que ce site est vulnérable à une attaque XSS stockée.



Etape : Réinitialisation pour un autre niveau

Nous avons donc réussi à exploiter un XSS stocké à faible niveau de sécurité. Maintenant, chaque fois que nous rafraîchissons la même page, nous obtenons cette boîte d'alerte car notre charge utile XSS est stockée dans le livre d'or. Si nous voulons exploiter cette vulnérabilité à un autre niveau de sécurité, nous devons d'abord effacer le livre d'or, sinon cette boîte d'alerte apparaîtra encore et encore. Avant de continuer, cliquez sur Clear Guestbook pour supprimer notre charge utile XSS du livre d'or



Etape : Changement de niveau et choix de vulnérabilité

Cliquez sur XSS (DOM) dans le volet de gauche pour sélectionner la vulnérabilité à DOM XSS.

Etape : Page de challenge

Nous sommes dans la page de challenge. Cliquez sur le bouton Select pour vérifier comment l'application se comporte.

Etape : Point d'entrée

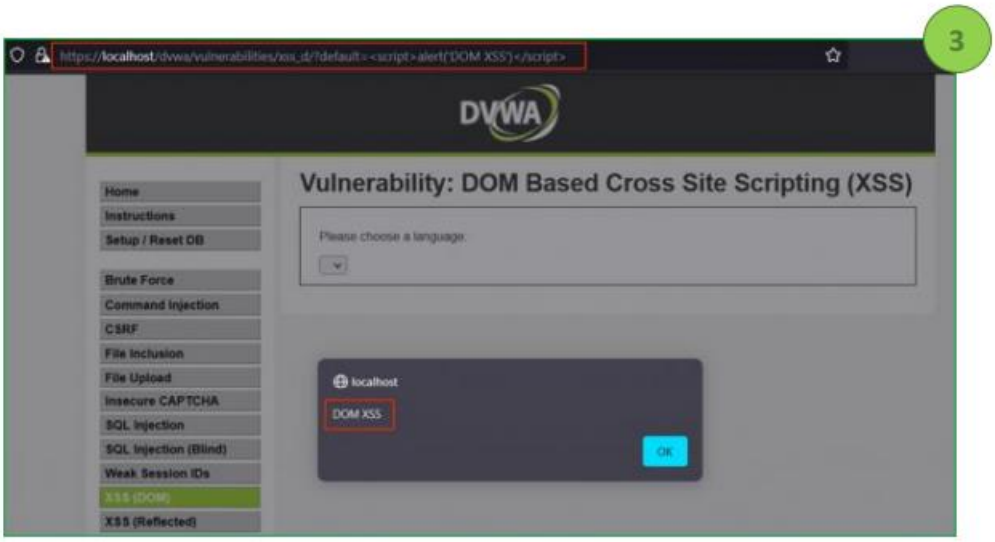
En cliquant sur le bouton, il définit la valeur du paramètre par défaut en anglais dans l'URL.

Comme nous le savons déjà, un paramètre dans une URL peut également être une source d'entrée. Modifions donc la valeur du paramètre «default » dans l'URL avec une chaîne unique et vérifions le code source.

Etape : Injection de code

Puisque notre chaîne unique est reflétée dans le DOM HTML, injectons notre charge utile XSS de base à la place de hello dans le paramètre par défaut. Nous

pouvons clairement voir dans la capture d'écran que notre charge utile injectée a été exécutée avec succès et que nous avons obtenu un pop-up XSS.



Etape : Vérification de l'injection

Nous pouvons vérifier qu'après une exécution réussie, notre charge utile est devenue une partie du DOM HTML.