LISV Laboratoire d'ingénierie des systèmes de Versailles
UVSQ université PARIS-SACLAY

Programmation sécurisée (Secure coding)
Examen TP 2

Halim Djerroud

révision 1.0

L'objectif de ce travail pratique est d'analyser les vulnérabilités d'une application web en mettant en pratique différentes attaques et en documentant leurs impacts.

1 Choix de l'Application

Pour réaliser ce TP, nous utiliserons **DVWA** (Damn Vulnerable Web App), une application web vulnérable écrite en PHP/MySQL. Cette application est légère, simple à utiliser et regorge de vulnérabilités exploitables à des fins pédagogiques.

Vous pouvez télécharger et installer DVWA en suivant le lien : https://github.com/digininja/DVWA

1.1 Installation de DVWA

1. Télécharger le code source depuis GitHub.
2. Installer un serveur web local (Apache, MySQL et PHP) via `XAMPP` ou `Docker`.
3. Configurer la base de données en modifiant le fichier `config.inc.php`.
4. Accéder à DVWA via le navigateur à l'adresse `http://localhost/DVWA`.
5. Modifier le niveau de sécurité pour faciliter ou complexifier les attaques.

(Il est possible d'utiliser une alternative, par exemple un conteneur docker) ou une autre méthode.

2 Travail Demandé

Vous devez réaliser et documenter les attaques suivantes :

1. **Brute Force** : Tester l'authentification par force brute.
2. **Command Execution** : Exécuter des commandes système via une vulnérabilité de l'application.
3. **CSRF (Cross-Site Request Forgery)** : Exploiter une faille permettant d'exécuter des actions à l'insu de l'utilisateur.
4. **File Inclusion** : Inclure un fichier malveillant dans l'application.
5. **SQL Injection** : Exploiter une vulnérabilité SQL pour récupérer des données sensibles.
6. **SQL Injection Blind** : Effectuer une injection SQL sans retour explicite d'erreur.
7. **Upload de fichiers** : Mettre en ligne un fichier malveillant et en exécuter le contenu.
8. **XSS Reflected** : Injecter du code JavaScript malveillant exécuté immédiatement.
9. **XSS Stored** : Injecter un script malveillant stocké et exécuté sur plusieurs sessions.

3 Méthodologie

Pour chaque attaque, suivre la démarche suivante :

1. Décrire la vulnérabilité exploitée.
2. Identifier les outils utilisés (ex : Burp Suite, OWASP ZAP, sqlmap, etc.).
3. Détailler les étapes pour exploiter la faille.
4. Capturer et documenter les résultats obtenus.
5. *Facultatif : Proposer des solutions pour corriger la vulnérabilité.*

4 Livrables

Les livrables à fournir à la fin du TP sont les suivants :
— **Rapport d'analyse de sécurité** généré par OWASP ZAP.
— **Captures d'écran** illustrant les attaques effectuées (injections SQL, XSS, vol de cookies, etc.).
— **Réponses aux questions d'évaluation** concernant chaque vulnérabilité exploitée.
— **Code et commandes utilisés** pour les attaques (scripts, requêtes SQL, payloads XSS, etc.).

5 Évaluation

Le rapport sera évalué en fonction des critères suivants :
— Exhaustivité et clarté des explications.
— Reproductibilité des attaques décrites.
— Documentation précise des résultats obtenus.
— *Facultatif : Propositions de solutions pour corriger les failles.*