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

Programmation sécurisée (Secure coding)
TP 3
Analyse de sécurité Web avec ZAP Proxy

Halim Djerroud

révision 1.0

L'objectif de ce TP est de permettre aux apprenants de :
- Comprendre le fonctionnement de ZAP Proxy.
- Intercepter, manipuler et rejouer des requêtes HTTP.
- Identifier les vulnérabilités suivantes :
  - Injections SQL
  - XSS (Cross-Site Scripting)
  - Fuite d'informations sensibles
  - Mauvaises configurations de sécurité
- Générer un rapport d'audit de sécurité.

Prérequis

- Connaissances de base des protocoles HTTP/HTTPS.
- Cours sur les vulnérabilités courantes du Top 10 OWASP.

Outils nécessaires

- ZAP Proxy (disponible sur https://www.zaproxy.org/).
- Navigateur web (Firefox ou Chrome) configuré pour utiliser un proxy.
- Application vulnérable (au choix) :
  - OWASP Juice Shop (https://github.com/juice-shop/juice-shop)
  - Damn Vulnerable Web Application (DVWA) (disponible via Docker).
  - BWAPP https://hub.docker.com/r/raesene/bwapp/

Exercice 1 : Installation et configuration de ZAP Proxy

1. **Installer ZAP Proxy** :
   - Téléchargez et installez ZAP Proxy à partir du site https://www.zaproxy.org/download/.
   - Lancez ZAP Proxy.
2. **Configurer le proxy du navigateur** :
   - Configurez Firefox ou Chrome pour utiliser le proxy 127.0.0.1 :8080.
   - Installez le certificat SSL de ZAP pour éviter les erreurs SSL (disponible sur http://zap).

Exercice 2 : Interception des requêtes HTTP

1. Accédez à l'application cible (ex : OWASP Juice Shop) via le navigateur.
2. Activez l'option "Break" dans ZAP Proxy.
3. Réalisez une action dans l'application (ex : connexion, recherche de produit, etc.).
4. Analysez la requête interceptée par ZAP Proxy :
   - Notez les champs de l'URL, les en-têtes HTTP et les cookies.
   - Essayez de manipuler les paramètres d'entrée (ID de produit, champs de recherche, etc.).

Exercice 3 : Analyse des vulnérabilités

1. Lancez un **scan actif** de l'application cible :
   — Faites un clic droit sur l'URL cible et sélectionnez "Active Scan".
   — Laissez le scan s'exécuter et surveillez les vulnérabilités détectées.
2. Analysez les résultats du scan :
   — Identifiez les vulnérabilités détectées par ZAP Proxy (XSS, injections SQL, etc.).
   — Notez les recommandations associées.

Exercice 4 : Exploitation manuelle des vulnérabilités

Test d'injection SQL

— Accédez au formulaire de connexion de l'application.
— Testez l'injection SQL suivante :

```
' OR '1'='1' --
```

— Notez si vous accédez au compte administrateur.

Test de XSS (Cross-Site Scripting)

— Recherchez un champ de saisie utilisateur (recherche de produit, commentaires, etc.).
— Testez l'injection suivante :

```
<script>alert('XSS');</script>
```

— Notez si le script est exécuté.

Test de fuite d'informations

— Recherchez les informations d'en-tête HTTP interceptées par ZAP.
— Recherchez la présence de cookies de session.
— Vérifiez si des informations sensibles (tokens, clés API) sont transmises.

Exercices pratiques

1. Expliquez les vulnérabilités découvertes dans le rapport d'analyse.
2. Identifiez au moins une injection SQL et une XSS.
3. Modifiez une requête interceptée et analysez la réponse du serveur.
4. Évaluez si des cookies sensibles sont marqués comme `HttpOnly` et `Secure`.

Questions d'évaluation

1. Quels types de vulnérabilités ZAP Proxy a-t-il détectés ?
2. Quelles actions recommanderiez-vous pour corriger les vulnérabilités identifiées ?
3. Comment éviter les attaques de XSS sur une application web ?
4. Quelles sont les bonnes pratiques de gestion des cookies de session ?
5. Que signifie "HttpOnly" et "Secure" pour un cookie ?

Exercice Avancé

— Automatisation des scans : Configurez un script Python pour lancer ZAP Proxy en ligne de commande.
— Intégration CI/CD : Intégrez ZAP Proxy dans une pipeline CI/CD (Jenkins, GitHub Actions, etc.).
— Exploitation avancée des sessions : Simulez une attaque de session hijacking (vol de session).

Livrables

Les livrables du TP incluent :

— **Rapport d'analyse de sécurité** généré par ZAP.
— **Captures d'écran** des tests d'injection SQL, XSS et de vol de cookies.
— **Réponses aux questions d'évaluation** listées dans la section précédente.