Pentest et injections SQL

Installer ZAP

- ZAP : https://github.com/zaproxy/zaproxy/wiki/Downloads
- Sur les ordis de l’école : prendre version crossplatform puis exécuter « ./zap.sh » dans le dossier racine

Installer ZAP

- Configurer Firefox/Chrome :
  - Préférences → Proxy Réseau → Config manuelle
  - Proxy HTTP et SSL : localhost port 8080
- Certificat SSL/TLS :
  - ZAP : Options → Certificats SSL dynamiques → Générer puis sauvegarder
  - Navigateur : Onglet Sécurité → Afficher les certificats → Autorités → Importer le certificat généré

Utiliser SQLmap

- http://sqlmap.org/ → télécharger le zip ou le tar.gz
- Se placer dans le dossier racine de l’appli
- « python sqlmap.py [commandes] »
- Ex : « python sqlmap.py -h » pour le help

Installer TheBodgeItStore

- ATTENTION : changer le port du proxy ZAP **avant ! (ex : 8000)**
- Télécharger Tomcat Apachehttps://tomcat.apache.org/download-80.cgi#8.5.28
- Télécharger BodgeIt 1.4 https://code.google.com/archive/p/bodgeit/downloads
- Extraire les deux dossiers dans Home ou Documents
- Déplacer bodgeit.war dans apache-tomcat/webapps
- Aller dans apache-tomcat/bin
- Exécuter startup.sh (linux) ou startup.bat (Windows)
- Dans Firefox : http://localhost:8080/bodgeit/

Premières analyses

- Se créer un compte dans bodgeit
- Dans ZAP, clic droit sur le dossier bodgeit → attaquer → indexer → indexer uniquement la sous-arborescence
- Premières observations ?

Configuration avancée de ZAP

- Clic droit sur le dossier bodgeit → inclure dans le contexte → contexte par défaut
- Exclure du contexte les pages register.jsp et logout.jsp
- Se déconnecter puis intercepter une requête de connexion → copier le contenu et l’adresse de la requête dans un fichier

Configuration avancée de ZAP

- Fichier → propriétés de la session → Contextes → Authentification → Authentification par formulaire → remplir avec les données copiées → remplir les paramètres de nom et de mot de passe avec les bonnes données

Analyses poussées

- Clic droit sur le dossier bodgeit → attaquer → Balayage actif avancé
- Patienter…
- Nouvelles observations ?

Utilisation de SQLmap

- Envoi d’une analyse :
- GET :

```
sqlmap -u "http://example.com/index.php?v1=a&v2=a"
```

- POST :

```
sqlmap -u "http://example.com.index.php" --data="v1=a1&v2=a2"
```

Utilisation de SQLmap

- Si vecteur d’attaque trouvé :
- Chercher nom des BDD :
  - sqlmap -u "http://example.com/index.php?v1=a&v2=a" --current-db
- Chercher tables :
  - sqlmap -u "http://example.com/index.php?v1=a&v2=a" --tables
- Dumper une table :
  - sqlmap -u "http://example.com/index.php?v1=a&v2=a" -T table --dump