OWASP Top 10 - Cheat Sheet (2021)

L'OWASP Top 10 est une classification des vulnérabilités de sécurité les plus critiques dans les applications web. Voici une liste complète avec des exemples d'attaques, des outils de test, et des commandes pour chaque type de vulnérabilité, basée sur la version 2021.

1. Contrôle d'accès non sécurisé (Broken Access Control)

Les utilisateurs peuvent accéder à des fonctions ou données non autorisées.

Exemple d'attaque :

```
curl -X GET "http://site.com/admin" -b "sessionid=xyz"
```

Outils de test :

- Burp Suite : Modification des requêtes HTTP pour contourner les contrôles d'accès.
- Fuzzing : Envoi de requêtes inattendues pour découvrir des failles.

2. Cryptographie Inadéquate (Cryptographic Failures)

Mauvaise protection des données sensibles par chiffrement insuffisant ou inapproprié.

Test avec Curl :

```
curl -X GET "http://site.com/api/userinfo" -H "Authorization: Bearer
<token>"
```

Outils de test :

- TestSSLServer : Analyse SSL/TLS d'un serveur web.
- OpenSSL : Test manuel de configurations SSL/TLS.

3. Injection

L'injection permet d'exécuter des commandes arbitraires sur un système cible.

Exemple d'attaque :

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

Outils de test :

```
sqlmap -u "http://site.com/login.php?id=1" --dump
```

4. Conception Insecure (Insecure Design)

Manque de contrôle de sécurité dès la phase de conception.

Outils de test :

- Threat Modeling : OWASP Threat Dragon.
- Analyse de conception par revue manuelle.

5. Mauvaise Configuration de Sécurité

Mauvaise gestion des permissions, services exposés inutilement.

Scan avec Nmap :

```
nmap -A -p- --script=vuln site.com
```

6. Vulnérabilité et intégrité des composants (Vulnerable and Outdated Components)

Utilisation de composants obsolètes ou vulnérables.

Scan de vulnérabilités :

```
nmap --script http-vuln* -p 80 site.com
```

Outils additionnels :

- Dependency-Check : Scanner des dépendances pour détecter les vulnérabilités connues.
- Retire.js : Détection de librairies JavaScript vulnérables.

7. Identification et Authentification Manquantes ou Défaillantes (Identification and Authentication Failures)

Problèmes dans la gestion de l'identité des utilisateurs.

Outils de test :

```
hydra -l admin -P /usr/share/wordlists/rockyou.txt http-post-form
"/login.php:username=^USER^&password=^PASS^:F=Incorrect"
```

8. Faille de Sécurité Logicielle (Software and Data Integrity Failures)

Absence de mécanismes pour vérifier l'intégrité du code et des données.

Outils de test :

- Hashcat : Vérification d'intégrité par hachage.
- Burp Suite : Manipulation de données pour tester l'intégrité.

9. Désérialisation Insecure

Exploitation d'objets sérialisés pour exécuter du code malveillant.

Exemple d'attaque PHP :

```
O:4:"Test":1:{s:4:"data";s:12:"Malicious Code";}
```

Outils de test :

```
ysoSerial <payload>
```

10. Surveillance et Journalisation Insuffisantes (Security Logging and Monitoring Failures)

Absence de logs permettant la détection d'attaques.

Outils de monitoring :

- Wazuh (SIEM Open Source)
- Splunk
- ELK Stack (Elasticsearch, Logstash, Kibana)

Exemple de configuration Elasticsearch :

```
curl -XGET 'localhost:9200/_cat/indices?v&pretty'
```