CHELLE Raymond 19/01/2026

Rapport de TP :

Sommaire :

I. Introduction

Dans le cadre de ce TP portant sur les **travaux en laboratoire informatique**, j'ai réalisé un audit de sécurité centré sur la robustesse des identifiants de connexion. L'objectif principal était de simuler une attaque réelle pour tester l'efficacité de la sensibilisation des utilisateurs aux bonnes pratiques de cybersécurité.

À travers la mise en place d'un environnement virtualisé combinant **Windows 10** (la cible) et **Kali Linux** (l'attaquant), j'ai mis en œuvre différentes techniques de récupération et de cassage de mots de passe. Ce rapport détaille la méthodologie employée, allant de l'extraction des hachages (hashes) système à l'utilisation d'outils professionnels tels que **John the Ripper** et **Ophcrack**, afin de démontrer la vulnérabilité des mots de passe trop simples face aux outils d'automatisation modernes.

II. Le travail à réaliser

1. Dans un premier temps, j'ai configuré mon environnement de test sur Windows 10. Afin de tester différentes résistances, j'ai créé trois comptes utilisateurs distincts via l'invite de commande : le compte 'ENEDIS' avec un mot de passe très court de 4 caractères, le compte 'MSA' avec une phrase secrète complexe de plus de 8 caractères

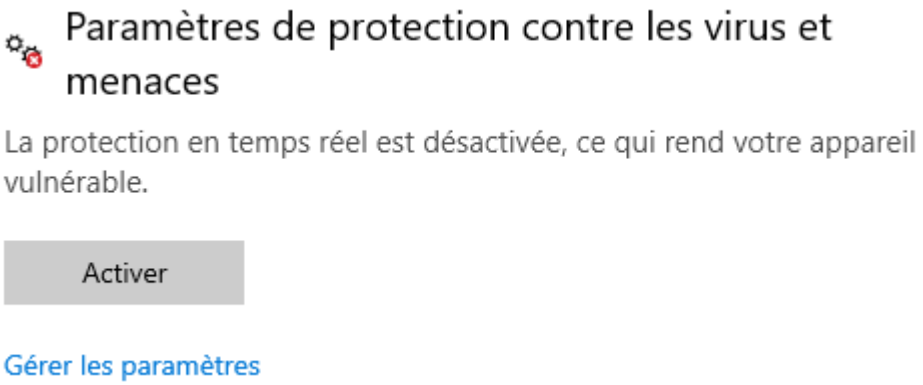
incluant des symboles, et enfin le compte 'CLIC'.

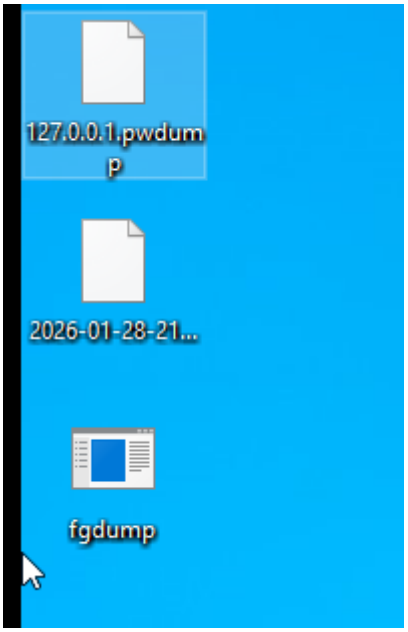
```
C:\Windows\system32>net user ENEDIS 1234 /add
```

```
C:\Windows\system32>net user MSA Soleil2026! /add
```

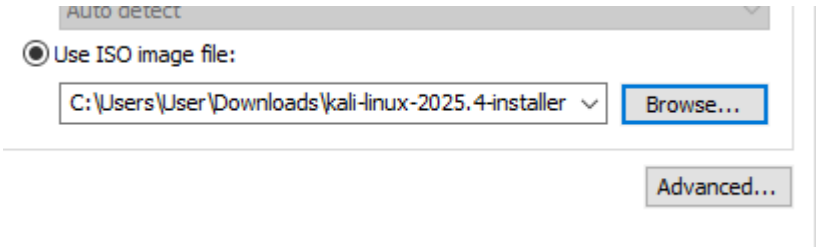
```
C:\Windows\system32>net user CLIC Chelle1234 /add
La commande s'est terminée correctement.
```

Pour pouvoir extraire les empreintes (hashes) de ces mots de passe, j'ai dû désactiver la protection en temps réel de Windows Defender, car l'outil FGDUMP que j'ai utilisé pour récupérer le fichier 127.0.0.1.pwdump est légitimement détecté comme une menace par le système.

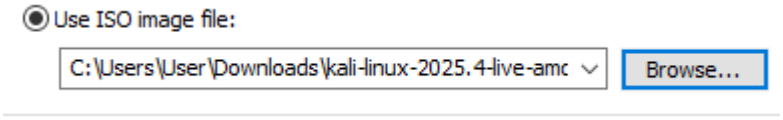




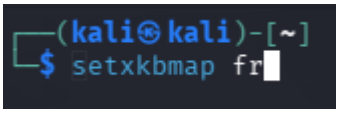
2.J'ai ensuite basculé sur la machine Kali Linux pour endosser le rôle de l'auditeur.



Je me suis en fait trompé, j'ai installé la version Installer au lieu de la version Live. Je suis donc allé sur le site officiel pour télécharger le Kali Live. J'ai eu un fichier .torrent, je suis donc aller sur utorrent afin de transformer ce fichier en iso.



Après avoir configuré mon clavier en AZERTY avec la commande setxkbmap fr



,j'ai dû monter la partition Windows de mon disque dur dans le répertoire /mnt.

```
└─$ sudo fdisk -l
Disk /dev/nvme0n1: 60 GiB, 64424509440 bytes, 125829120 sectors
Disk model: VMware Virtual NVMe Disk
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: gpt
Disk identifier: 789A7CFD-224F-4CE4-A582-6926A78D5384

Device             Start       End   Sectors  Size Type
/dev/nvme0n1p1      2048    206847    204800  100M EFI System
/dev/nvme0n1p2    206848    239615     32768   16M Microsoft reserved
/dev/nvme0n1p3    239616 124677874 124438259 59.3G Microsoft basic data
/dev/nvme0n1p4 124678144 125825023   1146880  560M Windows recovery environme
```

```
┌──(kali㉿kali)-[~]
└─$ sudo mount /dev/nvme0n1p3 /mnt

┌──(kali㉿kali)-[~]
└─$
```

Cela m'a permis de naviguer dans l'arborescence de fichiers de Windows alors que le système était éteint. J'ai ainsi pu récupérer le fichier de hashes (127.0.0.1.pwdump) que j'avais préparé précédemment et le copier sur mon bureau Linux pour commencer l'analyse.

3.

1. **Attaque par dictionnaire** (la plus rapide pour les mots de passe courants) : john --format=NT --wordlist=/usr/share/wordlists/rockyou.txt ~/Desktop/127.0.0.1.pwdump
2. **Attaque incrémentale** (Brute force pure si le dictionnaire échoue) : john --format=NT --incremental ~/Desktop/127.0.0.1.pwdump
3. **Afficher les résultats** : john --format=NT --show ~/Desktop/127.0.0.1.pwdump

"Pour casser les mots de passe, j'ai utilisé John the Ripper. J'ai spécifié le format '--format=NT' car Windows utilise le hachage NTLM. J'ai d'abord lancé une attaque par dictionnaire en utilisant le fichier 'rockyou.txt'. Cette méthode compare les hashes du fichier avec ceux de millions de mots de passe connus. Pour les comptes qui résistaient, comme 'MSA', j'ai utilisé le mode '--incremental'. Ce

mode teste mathématiquement toutes les combinaisons possibles de caractères. J'ai pu constater que le compte 'ENEDIS' a été craqué quasi instantanément, tandis que les mots de passe plus longs demandent un temps de calcul beaucoup plus important."

5.

En complément, j'ai utilisé l'outil graphique Ophcrack pour tester l'efficacité des 'Rainbow Tables'. J'ai chargé mon fichier de hashes via l'option 'Load PWDUMP'. Après avoir installé les tables 'vista_proba_free', j'ai lancé le processus. Contrairement à John qui calcule les hashes à la volée, Ophcrack recherche des correspondances dans des tables de données pré-calculées. J'ai remarqué que cette technique est extrêmement efficace pour retrouver des mots de passe alphanumériques simples, affichant en clair les mots de passe de mes utilisateurs en quelques secondes dans l'interface.

4 et 6

À l'issue de mes tests, j'ai constaté que la sécurité repose sur trois piliers : la **longueur** (minimum 12 à 14 caractères), la **complexité** (mélange de majuscules, minuscules, chiffres et caractères spéciaux) et l'**imprévisibilité** (ne pas utiliser de mots du dictionnaire ou d'informations personnelles). J'ai pu observer qu'un mot de passe simple est découvert en quelques secondes, alors qu'un mot de passe complexe rend l'attaque par dictionnaire totalement inefficace.

III. Conclusion

Ce travail pratique m'a permis de comprendre concrètement comment un auditeur de sécurité (ou une personne malveillante) peut usurper des identifiants en exploitant les faiblesses des politiques de mots de passe.

L'utilisation de **John the Ripper** m'a montré l'efficacité redoutable des attaques par dictionnaire et par force brute incrémentale, tandis que l'outil **Ophcrack** a illustré la rapidité des "Rainbow Tables" (tables

arc-en-ciel) pour les mots de passe simples. La leçon majeure de cet audit est sans appel : la sécurité d'un système ne repose pas uniquement sur ses logiciels, mais sur la rigueur des utilisateurs.

En conclusion, pour garantir une protection efficace, il est impératif d'adopter des mots de passe longs, complexes et imprévisibles. Ce TP souligne l'importance vitale d'informer les utilisateurs sur les risques encourus et de promouvoir des usages numériques responsables au sein de toute organisation. J’ai bien aimé faire ce TP, mais je l’ai trouvé compliqué honnêtement.